Symposium OPEN IVR
SNA Host Communications Guide

Product release: 4.0
Document release: Standard 1.0
Date: July 1998

Copyright © 1998 Northern Telecom
All rights reserved

Printed in the United States of America

Information is subject to change without notice. Northern Telecom reserves the right to make changes in design or components as progress in engineering and manufacturing may warrant.

Nortel, Symposium, IVR, and Meridian 1 are registered trademarks of Northern Telecom. APERTUS EXPRESS is a registered trademark of Computer Network Technology, Inc. Digi is a registered trademark of DIGI International. Motif is a trademark of Open Software Foundation Inc. SCO is a registered trademark of The Santa Cruz Operation, Inc. UNIX is a registered trademark of Novell, Inc. Voicetek and VTK are registered trademarks of Voicetek Corporation. X Window System and X are trademarks of the Massachusetts Institute of Technology.

Publication history

July 1998

This is the Standard 1.0 release of the *Symposium OPEN IVR SNA Host Communications Guide.*

Contents

About this guide xi
Who should use this guide . . . xi
Document organization . . . xi
Supporting documentation . . . xii
Conventions used in this guide . . . xiv

1 Host communication links 1-1
Symposium OPEN IVR platform . . . 1-2
Host communication hardware and software . . . 1-3
Creating Symposium OPEN IVR applications that communicate with an SNA host . . . 1-5
AP software host communication cells . . . 1-6
TRS . . . 1-8
TRSC . . . 1-8
TRSL . . . 1-8
EXPRESS communications software and hardware . . . 1-9
EHLLAPI . . . 1-9
Terminal emulation . . . 1-9

2 Host connectivity overview 2-1
Overview . . . 2-1
Operating system and supported software . . . 2-1
IBM 3270 emulation characteristics . . . 2-2
IBM 5250 emulation characteristics . . . 2-2
SNA communication terms . . . 2-2
3270/5250 SNA/SDLC card . . . 2-2
3270/5250 SNA/SDLC card—pinout and pin assignments . . . 2-4

Overview 2-4
DB-15 connectors and cabling 2-4
Installing the SYNC/570 adapter 2-7
3270/5250 Token Ring card 2-9
Installing the Madge ISA Token Ring card 2-11
Configuring the Madge ISA Token Ring card 2-13
Sample VTAM Gen 2-15

3 Installing and configuring EXPRESS 3270/5250 SNA/SDLC 3-1

Supporting documentation 3-2
Site planning 3-3
Reviewing physical configuration requirements 3-3
Verifying package contents 3-3
Installing the communications hardware 3-4
Completing the installation of the Madge ringnode adapter 3-5
Installing the EXPRESS software 3-6
Files modified by the EXPRESS installation 3-7
Beginning the EXPRESS SNA software installation 3-8
Removing earlier versions of the EXPRESS software 3-8
Installing the EXPRESS SNA software 3-9
Starting EXPRESS 3-15
Configuring EXPRESS 3-16
Defining links, stations, and logical units 3-16
Establishing a naming convention 3-16
Defining links and stations 3-17
Defining LU pools (3270 only) 3-18
Defining LU 6.2 (5250 only) 3-18
Defining remote LUs 3-19
Defining a mode 3-20
Defining users and sessions 3-20
Setting limits for terminal emulation 3-21
Verifying connections to the host computer 3-22
Checking the status of your links and stations 3-22
Testing 3270/5250 host communications 3-23
Displaying an interactive session 3-24
Starting up EXPRESS daemons and users 3-25
Editing system startup files 3-30
Modifying the /etc/rc2.d/S99zgen file 3-31

4 Installing and configuring EXPRESS TN3270 or TN5250 4-1

System requirements 4-2
Site planning. 4-2
Verifying required physical configurations 4-2
Verifying package contents 4-2
Installing EXPRESS TN3270 or TN5250 software. 4-3
Setting limits for terminal emulation 4-5
Starting EXPRESS TN3270 or TN5250 4-7
Starting the EXPRESS PS process 4-7
Configuring EXPRESS. 4-8
Defining users and session groups 4-8
Sample session file for TN5250 communications 4-9
Sample session file for TN3270 communications 4-12
Completing your configuration 4-14
Editing system startup files. 4-17
Starting user groups in the background 4-17

5 Configuring Symposium OPEN IVR host communications 5-1

Setting up the trs.conf file. 5-2
Setting up the trs.sdf file. 5-5
Specifying separate IDs and passwords for each LU. 5-6
Bringing up Symposium OPEN IVR, including TRS. 5-7
Using start-up switches 5-7
Completing the Symposium OPEN IVR host communication configuration 5-8
Mapping channels to host sessions 5-10
Sample configuration and template files 5-10

6 Creating template files 6-1

Overview of Action templates. 6-2
Purpose of Screen and Action templates. 6-2
Purpose and structure of the trs.conf file 6-5
Types of Action templates 6-6
Creating Action templates 6-8
Action template guidelines 6-8
Sample host application. 6-9
Initial-Action templates. 6-16

Action templates 6-17
Reset-Action templates 6-18
Logout-Action template 6-20
Keep-Alive or Heartbeat Action template 6-23
Creating Screen templates 6-24
Screen template syntax 6-28
Testing your Action and Screen templates 6-37
Using trs -c 6-37
Using the TRS log file 6-40

7 Using the COM cells to access the host computer 7-1
Overview of the application development process 7-2
Setting the COMI cell parameters 7-2
Setting the COMO cell parameters 7-5
Using the COMA cell 7-9
COMA cell parameters window branches 7-9
Determining successful COM cell communication 7-10
Sample application 7-10

8 Troubleshooting 8-1
Debugging the EXPRESS application 8-1
Log file 8-2
Trace file 8-2
Background process debugging 8-2
Traceable processes 8-3
Debugging the TRS application 8-5
Running TRS in verbose mode 8-5
TRS audit tool 8-5
Using the audit tool for dynamic logging 8-6
Changing log file size 8-6
Specifying logical units 8-7
Troubleshooting Action templates 8-8
Troubleshooting host communications 8-8

Appendix A: Planning your configuration **A-1**

Appendix B: Sample group host configuration **B-1**

Glossary **Glossary-1**

Index **Index-1**

About this guide

Who should use this guide

This *SNA Host Communications Guide* is intended to be used by Symposium OPEN IVR application developers whose voice applications require access to mainframe resources using the SNA communication protocols. In addition, it is assumed that, as a Symposium OPEN IVR application developer, you are familiar with SNA protocol-based applications and experienced in creating voice applications with Symposium OPEN IVR software. You should also be familiar with the UNIX operating system.

Document organization

This guide contains the following sections:

- This chapter, "About this guide", provides an overview of the guide.
- Chapter 1, "Host communication links", provides information about building upon the Symposium OPEN IVR telephony technology to create communications applications that interchange data with an SNA host.
- Chapter 2, "Host connectivity overview", provides information and instructions for host communication environments.
- Chapter 3, "Installing and configuring EXPRESS 3270/5250 SNA/SDLC", lists installation prerequisites and presents procedures for installing, starting, and configuring EXPRESS 3270 or 5250 software.
- Chapter 4, "Installing and configuring EXPRESS TN3270 or TN5250", lists installation prerequisites and presents procedures for installing, starting, and configuring EXPRESS TN3270 or TN5250 software.
- Chapter 5, "Configuring Symposium OPEN IVR host communications", describes the process of configuring the host link.

- Chapter 6, "Creating template files", provides information about using action and screen templates.
- Chapter 7, "Using the COM cells to access the host computer", provides information about setting the parameters for each of the COM cells and illustrates a sample application involving these cells.
- Chapter 8, "Troubleshooting", presents the tools that the Symposium OPEN IVR application developer can use to gather and debug data.
- Appendix A, "Planning your configuration", presents worksheets to assist you in configuring your system.
- Appendix B, "Sample group host configuration", shows some of the elements that may need to be configured on your system, and some sample settings.
- The Glossary provides definitions of terms used or related to the material in this guide.

Supporting documentation

The information supplied with your vendor's host connectivity package provides information necessary for hardware and software installation and configuration.

Apertus provides a complete documentation set that covers the installation and configuration of the APERTUS EXPRESS software package. Refer to the Apertus documentation for information about installing APERTUS EXPRESS software and to reference details of the SNA interface configuration and operation.

This guide refers to the following APERTUS EXPRESS manuals:

- *APERTUS EXPRESS Administrator's Guide*
- *APERTUS EXPRESS Desktop Configuration Guide*
- *APERTUS EXPRESS Installation Guide* (for your specific platform)
- *APERTUS EXPRESS Integrated Desktop Configuration Guide*
- *APERTUS EXPRESS New Features Guide*
- *APERTUS EXPRESS Release Notes*

- *APERTUS EXPRESS SNA Server Configuration Guide*
- *APERTUS EXPRESS TN3270/5250-X Administrator's Guide*
- *APERTUS EXPRESS User Interface Guide*

In addition, you may find it useful to have on hand the following Symposium OPEN IVR documentation:

- *Symposium OPEN IVR 4.0 System Administration Guide*
- *Symposium OPEN IVR 4.0 Installation, Upgrade, and Maintenance Guide*
- *Symposium OPEN IVR 4.0 Planning and Engineering Guide*
- *Symposium OPEN IVR 4.0 Application Development Guide*
- *Symposium OPEN IVR 4.0 Product Guide*

Conventions used in this guide

Throughout this guide, several typographic conventions are used to highlight certain types of information:

- Items that appear on the Symposium OPEN IVR screens are identified in quotes. For example:

  the “Timeout” part of the parameters window

- Prompts, file and menu names, directories, accounts, and options are shown in a different typeface. For example:

  `/sci>`

  `sci/install menu`

  `/u/nortel/3270/getbalance.act file`

- Symposium OPEN IVR buffer names are shown in all uppercase characters. For example:

  the CURRENT MESSAGE buffer

- Field names are shown in italics. For example:

  the *Device* field

- Items you must type are shown in bold in a different typeface. For example:

  type **`tn5250`** at the prompt

- Variables shown in command lines appear in italics. For example:

  the `host_cfg`*n* file, where *n* is a variable representing a board number

- Key names you press are shown in angled brackets. For example:

  the <F1> key

- Keyboards usually have keys named <Return> or <Enter> that perform the same function. For convenience, this guide uses the keyname <Enter> to represent both keynames.

Chapter 1: Host communication links

This chapter shows the process of building upon Symposium OPEN IVR telephony technology to create communications applications that interact with a System Network Architecture (SNA) host. This chapter contains the following topics:

- host communication hardware and software
- creating Symposium OPEN IVR applications that communicate with an SNA host
- IVR Developer COM cells
- TRS
- TRSC
- TRSL
- EXPRESS communications software and hardware

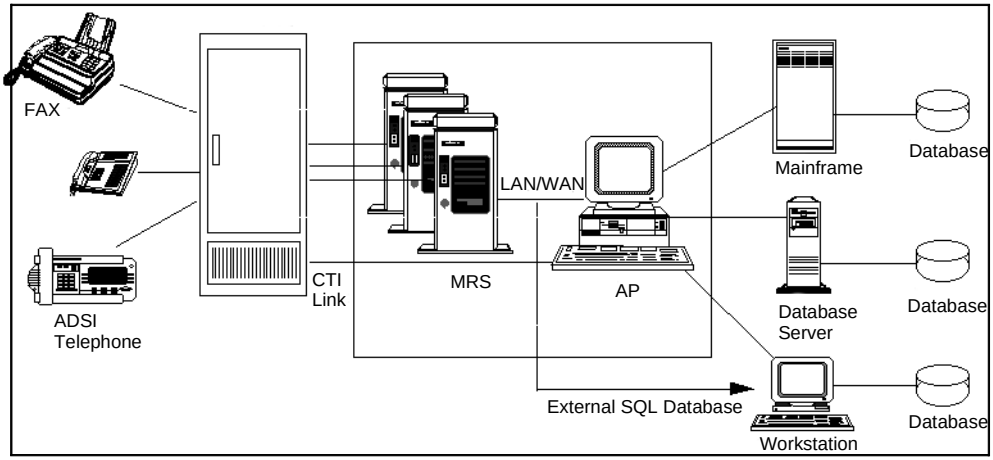
Symposium OPEN IVR SNA host link communication applications enable users to use a telephone to access information that resides on a mainframe host. You can create Symposium OPEN IVR applications that automatically and transparently transfer data between a mainframe host and a Nortel application processor.

Note: Only one host connectivity option is supported on your system.

Symposium OPEN IVR platform

The following figure illustrates the flow of data between a mainframe host and a Nortel application processor.

Figure 1-1: Symposium OPEN IVR platform



Host communication hardware and software

You can develop a Symposium OPEN IVR application that logs on to the host, manipulates the host application, and then stores the desired information in the buffers of a Symposium OPEN IVR application. You can install the host communication software on application servers that are running Symposium OPEN IVR software, Release 4.0. Refer to the *Symposium OPEN IVR Product Guide* for additional information about host communication support.

The application processor must contain the following components:

- either an SNA protocol connection to the mainframe host using an SDLC or a Token Ring connection

  or

  a TN protocol connection using TCP/IP over an Ethernet network
- Symposium OPEN IVR modules:
  - Terminal Resource Server (TRS)
  - Terminal Resource Server Client (TRSC)

You can configure your system to run 3270 or 5250 emulation software over either a TCP/IP protocol over an Ethernet network, or an SDLC or Token Ring network connection to the host mainframe.

Figure 1-2 displays a sample SNA host communication configuration using an SDLC or Token Ring network connection.

Figure 1-2: Sample SNA host communication configuration

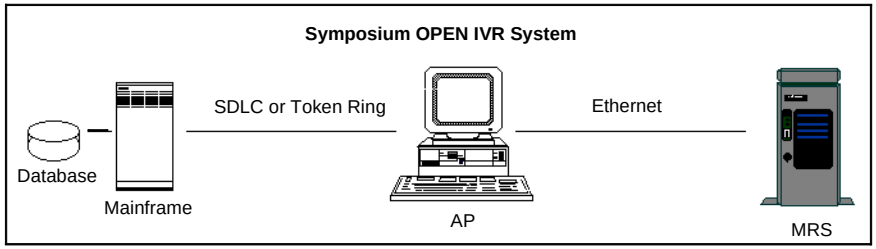
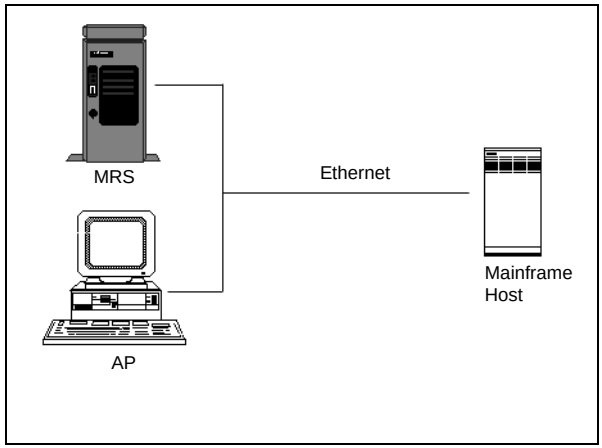


Figure 1-3 displays a sample TN host communication configuration using TCP/IP protocol over an Ethernet network.

Figure 1-3: Sample TN host communication configuration

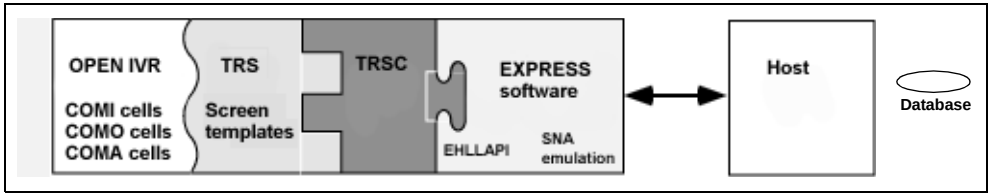


Creating Symposium OPEN IVR applications that communicate with an SNA host

Symposium OPEN IVR applications communicate with a mainframe host by establishing connections, and sending and receiving data over SNA sessions. This section explains the Symposium OPEN IVR host link software architecture.

The following compares the Symposium OPEN IVR architectural flow of information to pieces of a puzzle.This flow begins with a Symposium OPEN IVR application. The AP software application uses COMI and COMO cells to pass information to and from the mainframe host through Apertus EXPRESS software. The exchange of data from the AP software host communication cells occurs through the TRS modules.

Figure 1-4: Symposium OPEN IVR SNA gateway software architecture



ATTENTION

Since the SNA host link process is determined by the host environment, the host's system administrator needs to be aware of the SNA protocol that is being used so that the EXPRESS software can be configured properly.

AP software host communication cells

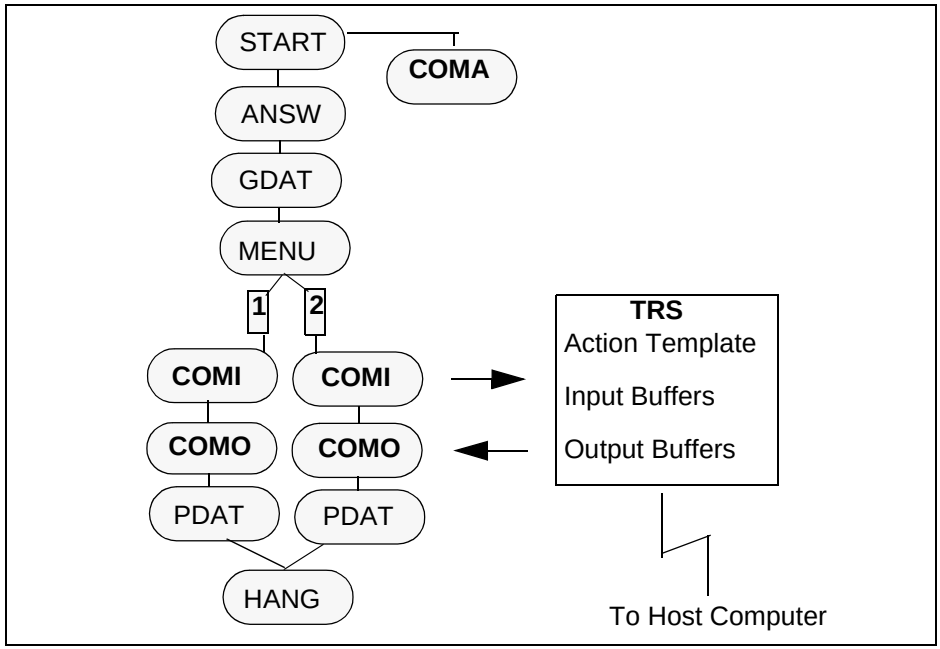
Symposium OPEN IVR applications use host communication cells to access a mainframe host. You can use the following host communication cells to communicate and exchange data with a mainframe host:

- COMI
- COMO
- COMA

Sample application call flow

The following figure illustrates an application call flow involving the COMA, COMI, and COMO cells.

Figure 1-5: The COM cells



Explanation of the sample application call flow

The COMI cell initiates data exchange by invoking an Action template. An Action template contains a list of screens that are traversed to complete the transaction. Different Action templates handle different contingencies in host transactions. Refer to Chapter 6, "Creating template files", for more information.

> ***Note:*** A transaction consists of the steps a terminal operator performs in order to complete a task. This includes logging in to the host, navigating through the appropriate screens, and logging out.

The COMO cell collects host data and places this retrieved information into application buffers. Any Symposium OPEN IVR cell can then use the information.

The COMA cell cancels or resets a host transaction. Typically, the COMA cell is executed to end the session or if there is a problem with the host transaction.

To use TRS, place the COM cells in the Symposium OPEN IVR application call flow at the point where you need to set up a host transaction or data exchange. For more information about the Symposium OPEN IVR COM cells, refer to Chapter 7, "Using the COM cells to access the host computer."

TRS

A Symposium OPEN IVR process called the terminal resource server (TRS) controls all SNA host link sessions and interfaces with the installed terminal emulation package. In addition, TRS manages all host connections in conjunction with Apertus software. The TRS runs as a stand-alone process within the Symposium OPEN IVR architecture and starts automatically when Symposium OPEN IVR is started.

The TRS process uses template files to control the communication process between the application and the host computer. The template files provide a way of understanding the screen images in a host application. These files reflect the location of the input and output fields within host applications. Refer to Chapter 6, "Creating template files", for more information about creating screen templates.

TRSC

The terminal resource server client (TRSC) communicates with the specific vendor communication package. The TRSC receives data from the TRS and formats it for use with the specific vendor hardware and software.

TRSL

The Terminal Resource Server Logger (TRSL) is the log message manager for the TRS and TRSC processes. When in verbose mode, the TRS and TRSC processes send log messages to the TRSL, which then adds these messages to the `trs.log` file. You can use the `trs.log` file to troubleshoot problems associated with the TRS and TRSC processes.

EXPRESS communications software and hardware

The Apertus EXPRESS communication multiprotocol adapter board and EXPRESS communications software manage the actual host connections that transmit and receive data complying with the SNA network protocols.

EHLLAPI

Using the IBM-defined Extended High Level Language Applications Programming Interface (EHLLAPI), you can search through the host screen data (also referred to as screen scraping) for specific strings of text, wait for specified actions, and perform other functions. You set up these transparent functions in the screen templates.

Terminal emulation

Terminal emulation occurs at the remote sites using standard SNA protocol terminal emulators. These terminals give multiple users access to resources residing on a host computer. The terminals use the specific protocol (3270 or 5250) for displaying information on their screens and returning screen contents to the IBM host. Each field on an input screen contains unique attribute information. Attributes identify the characteristics of the information written to the screen, such as highlighted text and reverse video.

The SNA communication adapter uses the same technology. An adapter installed in an application processor can emulate many concurrent SNA protocol terminal sessions over a single SNA connection. By complying with the SNA communication protocols, Symposium OPEN IVR applications can support multiple logical SNA terminal sessions. Physical terminals are not required. Your Symposium OPEN IVR applications can access the same host resources as operators using SNA protocol terminals.

For example, consider a menu-driven program on a host. This program allows an operator to enter and retrieve information stored in a database. Instead of connecting a terminal to the host and retrieving information manually, you can develop a Symposium OPEN IVR application that logs on to the host, manipulates the host application, and then stores the desired information in the buffers of a Symposium OPEN IVR application. Buffers can then be played to a person using the phone.

Optimally, your application can simultaneously run multiple terminal sessions and process different transactions for different users.

Chapter 2: Host connectivity overview

Overview

Host connectivity options provide an IVR application running on the AP with the ability to interface with several types of host computers over a number of different topologies. Symposium OPEN IVR supports the following host communication environments:

- IBM 3270 via SNA over SDLC
- IBM 3270 via SNA over Token Ring
- IBM 3270 via TCP/IP over Ethernet
- IBM 5250 via SNA over SDLC
- IBM 5250 via SNA over Token Ring
- IBM 5250 via TCP/IP over Ethernet
- VT100 via Async (over RS-232)

The IVR system emulates 3270 and 5250 through the use of EXPRESS software from Apertus Technologies.

The IVR system supports VT100 emulation through the use of software from DIGI International.

Operating system and supported software

Symposium OPEN IVR 4.0 uses the SCO OPENServer 5.0.4c Enterprise operating system.

For TCP/IP, it uses EXPRESS TN3270/5250 Release 1.31 SCN 5152.

For SNA, it uses EXPRESS Release 2.1 SCN 4153.

IBM 3270 emulation characteristics

EXPRESS provides full emulation of IBM 3174 cluster controllers and their attached 3178/3179 terminals for access to host environments such as MVS, VM, and VSE. EXPRESS emulates IBM 3270 display station models 1, 2, 3, 4, and 5 and provides screen presentation spaces ranging from 12 rows by 40 columns to 27 rows by 132 columns. It also supports seven colors, extended highlighting features, and full status display. EXPRESS uses LU1, LU2, and LU3 in support of 3270 emulation.

IBM 5250 emulation characteristics

EXPRESS emulates the IBM 5250 family of terminal devices to provide connectivity to mid-range systems such as the AS/400, and to other host systems that use 5250 communications. The IBM 5250 family of devices includes display stations that communicate with a host system through a cluster controller. The 5250 emulation supports seven colors, with screen presentation spaces of 24 rows by 80 columns, or 27 rows by 132 columns, and also supports extended highlighting features. The 5250 emulation emulates a 5394 Cluster Controller with 5251 Mod 2 terminals. The emulation uses 5250 passthrough and the 5250 terminal emulation uses APPC. EXPRESS uses LU 6.2 in support of 5250 emulation and CPI-C. APPC and LU6.2 are synonymous terms.

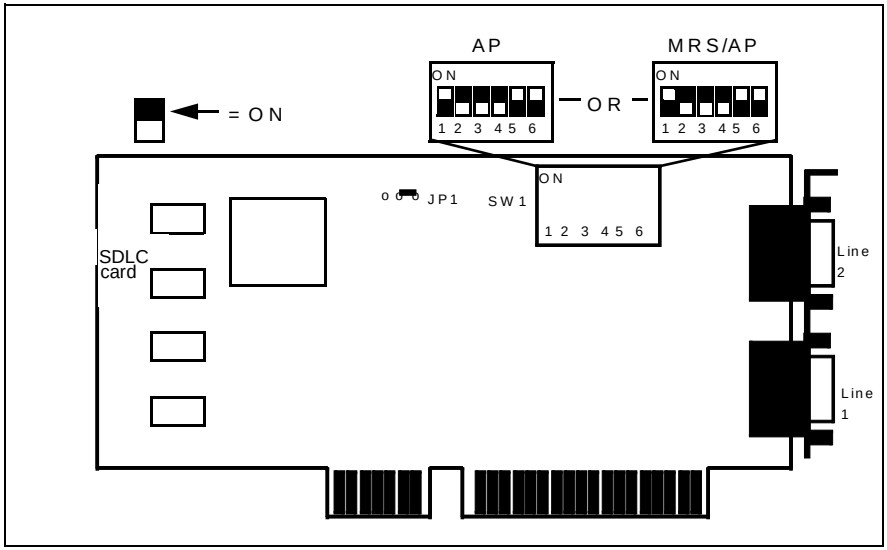
SNA communication terms

The configuration of the software for 3270 and 5250 requires the understanding of some basic communication terms, as described in the Glossary.

3270/5250 SNA/SDLC card

The SYNC/570 card is used for connectivity to the host using SNA/SDLC. The SYNC/570 is a high-performance expansion board which connects into an ISA expansion slot and contains two communication ports. Included with the SYNC/570 adapter are two cables (9-pin male to 25-pin male) for attaching to line 1 and 2 of the adapter. Also included is a loopback connector for diagnostics. Each of the two ports of the SYNC/570 has a high-density DB-15 connector to connect any device with an EIA-232-D interface to the computer. The board operates at line speeds from 4800 baud to more than 115 K.

Figure 2-1: SDLC card



3270/5250 SNA/SDLC card—pinout and pin assignments

Overview

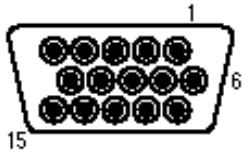
The SYNC/570 has two high-density 15-pin D-subminiature female port connectors. The connectors provide an EIA 232D interface and serial data transmission and reception.

Your application may specify the cable configuration required for connecting modems to a computer. If so, you should follow the recommendations. If cable configuration is not specified, you can use the cable types discussed below. Cables for use with synchronous modems need a minimum of eleven wires plus a shield.

DB-15 connectors and cabling

The DB-15 connectors are compatible with EIA-232-D devices. Figure 2-2, “DB-15 female connector and connector pin assignments,” shows the pinout and pin assignments for the DB-15 connectors on the SYNC/570 card.

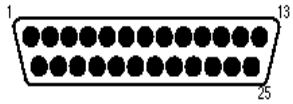
Figure 2-2: DB-15 female connector and connector pin assignments



1	Not connected
2	Transmitted data (TXD) Output
3	Received Data (RXD) Input
4	Request to Send (RTS) Output
5	Clear to Send (CTS) Input
6	Data Set Ready (DSR) Input
7	Signal Ground
8	Data Carrier Detect (DCD) Input
12	Receive Clock (RXC) Input
13	Transmit Clock (TXC) Output
14	Transmit Clock (TXC) Input
15	Data Terminal Ready (DTR) Output
shell	CGND

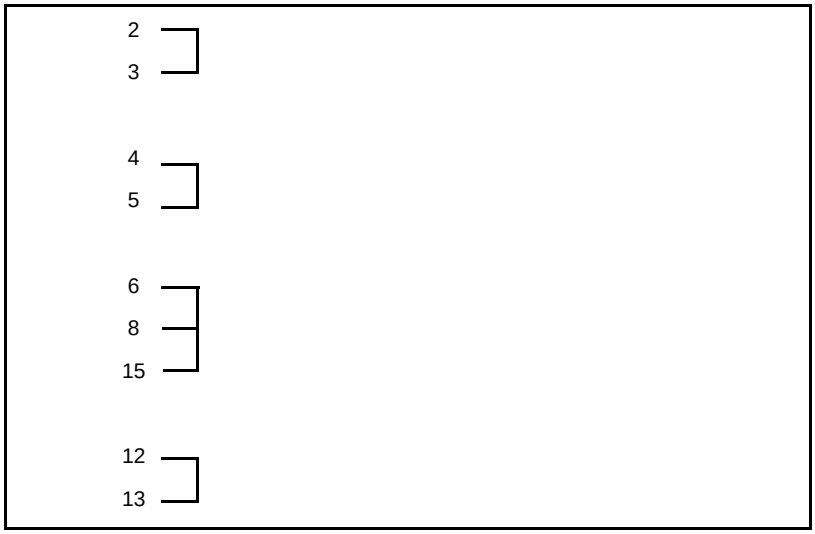
The cables included in the SYNC/570 package have a DB-15 male connector on the board end, and a DB-25 male connector on the peripheral end. The pinout and pin assignments for the DB-25 connector are shown in Figure 2-3, “Cable D-25 male connector and connector pin assignments.” The male loopback connector wiring diagram is shown in Figure 2-4, “SYNC/570 DB-15 male loopback connector wiring diagram.”

Figure 2-3: Cable D-25 male connector and connector pin assignments



1	Not connected
2	Transmitted data (TXD) Output
3	Received Data (RXD) Input
4	Request to Send (RTS) Output
5	Clear to Send (CTS) Input
6	Data Set Ready (DSR) Input
7	Signal Ground
8	Data Carrier Detect (DCD) Input
15	Transmit Clock (TXC) Input
17	Receive Clock (RXC) Input
20	Data Terminal Ready (DTR)Output
24	Transmit Clock (TXC) Output
shell	CGND

Figure 2-4: SYNC/570 DB-15 male loopback connector wiring diagram



Installing the SYNC/570 adapter

To install the SYNC/570 adapter, follow these steps.

CAUTION

Risk of data loss

When installing the SYNC/570 adapter, avoid damage from electrostatic discharge by using a grounding strap, by handling the board by the edges only, and by keeping the board in its protective package until you are ready to install it.

1 Shut down the AP or MRS/AP by following the established shutdown procedures.

2 Turn off power to the system, and disconnect the power cord.

3 Remove the side cover.

4 In an AP attached to an MRS/Tower or MRS/Rack (that is, not an MRS/AP), the board is installed in EISA slot 3.

Note: EISA slot 1 is at the bottom of the motherboard, slot 2 is above slot 1, and so on).

In an MRS/AP, the board is installed in EISA slot 1.

Note: Slot 6 can accept a board only if PCI slot 1 is empty (slot 3 is the top PCI slot).

5 Remove the slot cover, and save the screw for later use.

6 On an AP (not an MRS/AP), set the Switch Block 1 (SW1) on the SYNC/570 adapter to use I/O address 310 by setting the DIP switches to the following:

switch 1 -	off	switch 2 -	on
switch 3 -	on	switch 4 -	on
switch 5 -	off	switch 6 -	off

On an MRS/AP (not an AP), set the Switch Block 1 (SW1) on the SYNC/570 adapter to use I/O address 310 by setting the DIP switches to the following:

switch 1 -	off	switch 2 -	off
switch 3 -	on	switch 4 -	on
switch 5 -	off	switch 6 -	off

7 Firmly push the SYNC/570 card into the connector of the available slot. Do not force the adapter into the slot; it should slide in easily and if it does not, it may be misaligned. Secure the adapter with the screw removed in step 5.

8 Reinstall the side cover.

9 Two (2) cables and a loopback connector are shipped with the adapter. Connect one cable to Line 1 of the adapter. If there are two connections to the host, also connect the second cable to Line 2. If there is only one connection to the host, connect the loopback connector to Line 2.

10 Reconnect the power cord, and power up the AP.

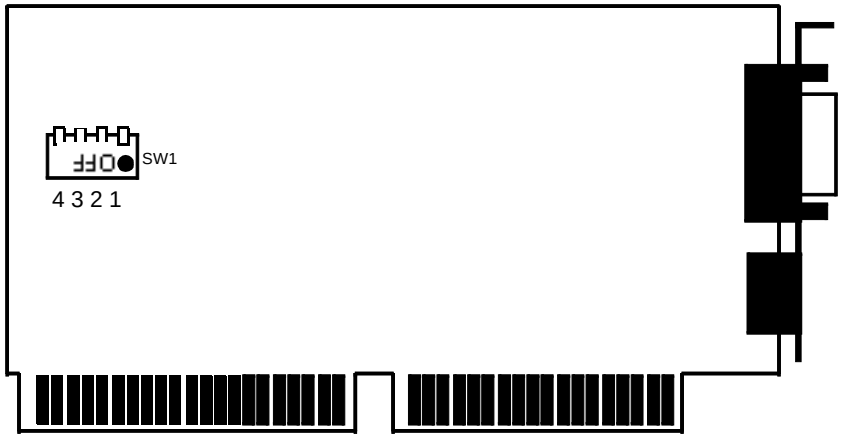
11 Connect the cables from the adapter to the host modem or null modem cable.

The hardware installation is complete.

3270/5250 Token Ring card

The Madge Smart 16/4 AT Plus Ringnode card is used for connectivity to the host using Token Ring.

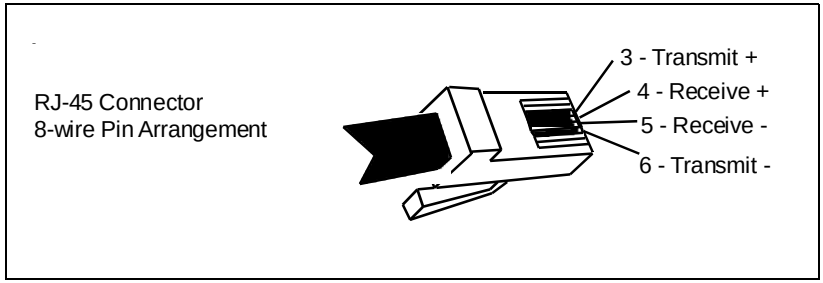
Figure 2-5: Madge Smart 16/4 AT Plus Ringnode Token Ring host card



Note: Set the SW1 switches to the following:.

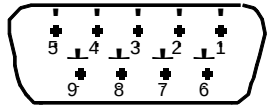
1 - ON
2 - ON
3 - OFF
4 - ON for 16 Mbps, OFF for 4 Mbps
Set the I/O address to 3A20

Figure 2-6: Token Ring pin arrangements



9-Pin connector pinouts

The 9-pin “D” subminiature STP cable pinouts on the Network Interface card consist of the following.



PIN	CABLE	Signal
1	Red	Receive +
5	Black	Transmit -
6	Green	Receive -
9	Orange	Transmit +

Pins 2, 3, 4, 7, and 8 are not used. The connector shield grounds the chassis.

Installing the Madge ISA Token Ring card

Note: Settings shown in this procedure are for the Madge Token Ring card used for host connectivity. If you are using the card for a LAN connection, refer to the installation procedure and settings in the *Symposium OPEN IVR Installation, Upgrade, and Maintenance Guide*

To install a Madge ISA Token Ring card, follow these steps..

DANGER

Risk of electric shock

High current inside the chassis can cause severe injury.

WARNING

Risk of equipment damage

Take precautions to protect internal components. Electrostatic discharge (ESD) can damage boards damaged or make them unusable. Wear an ESD wrist strap.

1 Ensure that you have performed a full backup of the server you are working on.

Refer to “Creating a system image backup” in the *Symposium OPEN IVR Installation, Upgrade, and Maintenance Guide*.

2 Shut down and power off the unit you are working on.

Refer to “Shutting down the AP or MRS/AP” and “Shutting down the MRS” in the *Symposium OPEN IVR Installation, Upgrade, and Maintenance Guide*.

3 Unplug the power cord for the unit from the electrical outlet.

4 Remove the cover from the unit.

Refer to Chapter 21, “Replacing hardware components”, for Release 4.0 hardware in the *Symposium OPEN IVR Installation, Upgrade, and Maintenance Guide*.

5 Clip the lead from your ESD wrist strap to an unpainted metal section of the chassis.

6 If you are replacing a Madge ISA Token Ring card, identify the slot that contains the faulty card. If you are adding a Madge ISA Token Ring card, select an empty ISA slot for the card.

7 Remove the slot’s retaining screw. If the slot is empty, remove the slot cover. If you are replacing a card in this slot, remove the card.

8 For host connectivity, set the switches to the following.

Network speed	**1**	**2**	**3**	**4**
4 Mbps	ON	ON	OFF	OFF
16 Mbps	ON	ON	OFF	ON

9 Install the Madge ISA Token Ring card.

Use the slot’s card guides as you insert the card edge connector into the slot. Press firmly until the card is securely seated in the slot.

10 Replace and tighten the retaining screw to secure the card.

11 Reinstall the cover for the unit.

Refer to “Replacing hardware components” in the *Symposium OPEN IVR Installation, Upgrade, and Maintenance Guide*.

12 Connect the LAN cable into the DB9 or RJ-45 connector (cable provided by customer).

13 Reconnect the power cord.

Continue with the steps in “Configuring the Madge ISA Token Ring card”.

Configuring the Madge ISA Token Ring card

1. Insert an MS-DOS 5.0 or higher version system disk in the floppy drive.
2. Power up the system.
3. At the DOS prompt, type **`c:`** and press <Enter>.
4. Type **`diag`** and press <Enter>.
5. Type **`4`**.

   The Madge Wizard Menu is displayed.
6. Tab to `Change Ringnode configuration`, and press <Enter>. The selected item has dashes around the text label.

   Wait for the Hardware Wizard to initialize before continuing.
7. Tab to `Custom`, and press <Enter>.

   Wait for the Hardware Wizard to initialize before continuing.
8. Tab to the window that lists the `Global system resources`.
9. Use the arrow keys to select the Smart 16/4 AT Plus Ringnode.
10. Tab to `Select`, and press <Enter>.
11. Tab to `Configure`, and press <Enter>.

    Wait for the utility to initialize data before continuing.
12. Configure the Madge ISA Token Ring card to the appropriate settings.

    Type the following values:

    I/O address: **`0x3A20 - 0x3A2F`**

    DMA channel: **`5`**

    Interrupt: **`15`**
    (*Note:* For the 700t MRS/AP, use IRQ 10 if the system contains an Aculab E1 card.)

    Ringspeed: **`4`** or **`16`**, depending on the hardware settings on the card

    Memory: **`Disabled`**

    Bus timing: **`Normal`**

    Bus mode: **`Async`**

    Transfer width: **`16`**
13. Smartrom and PIO Force should remain unchecked (disabled).

14 Tab to `Program`, and press <Enter>.

Wait while the utility updates the configuration.

15 A window is displayed with the following confirmation message:

```
The operation completed successfully.
```

16 Tab to `OK`, and press <Enter>.

17 Exit from all windows, and exit from the utility.

18 Type **q** to quit from the diagnostics menu.

19 Remove the MS-DOS bootup disk, and store it in a safe place.

20 Power off the system, and then power on.

Allow the system to boot normally.

21 To continue with the 3270/5250 SNA/SDLC installation, see Chapter 3, “Installing and configuring EXPRESS 3270/5250 SNA/SDLC”.

To continue with the TN3270/TN5250 installation, see Chapter 4, “Installing and configuring EXPRESS TN3270 or TN5250”.

Sample VTAM Gen

The following is a sample VTAM Gen on a mainframe. This sample can be used when you are attempting to obtain the correct parameters from the host to configure the EXPRESS software.

```
PCCU CUADDR=AA7,

   DUMPDS=DUMP,
   MAXDATA=5000
   SUBAREA=1,
   NETID=OREGON,          <-- Remote NET ID
   OWNER=MVSXA,           <-- Remote CONTROL POINT

.
. (many more lines here)
.

   DLOGMOD=D4C32782,
   SSCPFM=USSSCS,
   MAXDATA=265,
   MAXOUT=7
   PASSLIM=7,
   PACING=0,
   USSTAB=USSTABST,
   VPACING=0

LU LOCADDR=2
LU LOCADDR=3, DLOGMOD=DSC2K,USSTAB=U5STABSP
LU LOCADDR=4
LU LOCADDR=5
LU LOCADDR=7, DLOGMOD=DSC2K,USSTAB=USSTABSP
*************************************************
*
*  LAA   (MODEM ELIMINATOR) IVR
*
*************************************************

LAA    LINE ADDRESS=(021,FULL),

   MAXPU=2
   DUPLEX=FULL,
   SPEED=9600,            <-- Baud rate
   NEWSYNC=YES,           <-- Role=Negotiable
   RETRIES=(3,3,10)
```

```
SERVICE ORDER=(CAA1),MAXLIST=2
************************************************
*  CAA1   INTERACTIVE VOICE RECOGNITION
************************************************
CAA1   PU PUTYPE=2,
   ADDR=C1,
   MODETAB=LSTINCLM,
   DLOGMOD=D4C32782,
   SSCPFM=USSSCS,
   MAXDATA=265,<
   MAXOUT=7,
   PASSLIM=7,
   PACING=0,
   USSTAB=USSTABST,
   VPACING=0

LU LOCADDR=2
LU LOCADDR=3
LU LOCADDR=4
LU LOCADDR=5
LU LOCADDR=6
.
.
.
LU LOCADDR=30
LU LOCADDR=31
LU LOCADDR=32
LU LOCADDR=33
```

CAA1=Control Point

Secondary Station Address

Maximum Frame Size

Sequencing

Sequencing

Chapter 3: Installing and configuring EXPRESS 3270/5250 SNA/SDLC

This chapter discusses the process of setting up the EXPRESS 3270/5250 server software and hardware for host link communications. The EXPRESS SNA server supports Synchronous Data Link Control (SDLC) and Token Ring communications to mainframe host computers using either the 3270 or the 5250 protocol.

> ***Note 1:*** If you are using an Ethernet board for host connectivity, you can skip this chapter and continue with Chapter 4, "Installing and configuring EXPRESS TN3270 or TN5250".
>
> ***Note 2:*** EXPRESS is a product of Apertus, which has a complete documentation set that covers the installation and configuration of the EXPRESS software. Refer to the EXPRESS documentation that accompanies your software package for specific information about installing and configuring EXPRESS.

The sections of this chapter are presented in the order you follow to successfully establish the communications:

- site planning
- installing the communications hardware
- installing the EXPRESS software
- starting EXPRESS
- configuring EXPRESS

- verifying connections to the host computer

  This section outlines the steps to take and details related to the installation and configuration of the EXPRESS SNA server that are specific to Symposium OPEN IVR software.

Supporting documentation

You need to refer to the following documents to install the EXPRESS software:

- *EXPRESS Installation Guide*
- *EXPRESS Release Notes*

You need the following EXPRESS guides to reference details of the SNA interface configuration and operation:

- *EXPRESS Administrator's Guide*
- *EXPRESS Integrated Desktop Configuration Guide*
- *EXPRESS New Features Guide*
- *EXPRESS SNA Server Configuration Guide*
- *EXPRESS User Interface Guide*

Site planning

Verify your physical configuration requirements and inspect the package contents before you install and configure your system, as described in this chapter.

Reviewing physical configuration requirements

Before you start to install and configure your system, review the required physical configuration of the installation site:

- communications protocols
- number of logical units (LUs)
- communication media (SDLC or Token Ring)
- host information
  - LUs created/working
  - ports created/working
  - host MAC address (Token Ring)
  - secondary station address (SDLC)
  - network ID
  - XID block
  - control point

Verifying package contents

Before you start to install the software, verify that you received the package that you ordered, including the following items:

- correct number of logical units
- correct EXPRESS software version
- documentation

Installing the communications hardware

The communications adapter provides the physical connection between your application processor and the SDLC or Token Ring network. You install the communication adapter in an expansion slot in your application processor.

The communication hardware installation procedure consists of these steps:

- installing the communications adapter in your application processor
- connecting the communications adapter to your physical SDLC or Token Ring communications media
- verifying that the hardware is correctly integrated with your application processor

  Note: For this procedure, refer to the documentation for the specific communication adapter you are installing.

The SCO platform requires the following:

- interrupt level (IRQ)
- I/O address
- memory address
- DMA channel (Token Ring)

Completing the installation of the Madge ringnode adapter

Note: This procedure is for an SNA installation only. For Ethernet or TCP/IP installations, the cards are already installed.

To complete the installation of the Madge ringnode adapter, follow these steps.

1 Test the configuration from the main menu by selecting `F1 - Run Diagnostics`.
When the next menu appears, select `Standard Diagnostics`.

Note: The card must be attached to the Ring for the test to succeed.

2 Remove the disk, and press the Reset button to reboot the system from UNIX.

The hardware installation is complete.

Installing the EXPRESS software

EXPRESS is an integral part of the host link. The EXPRESS software package, coupled with a communications adapter, handles all the lower-level SNA communication. When this package is correctly installed and configured, the software transparently establishes and maintains an SDLC or Token Ring connection with the host.

The EXPRESS software package consists of the following components:

- EXPRESS SNA server
- EXPRESS integrated desktop
- EXPRESS API software development kit
- EXPRESS API runtime

Of these, you install only the EXPRESS SNA server and the EXPRESS integrated desktop. These components are essential for SNA communication. The server performs the actual terminal emulation, and the desktop is the user interface to the server. You use the desktop to configure the server.

EXPRESS installs the desktop automatically when you install the SNA server. During installation, the EXPRESS installation script refers to the server and desktop as one package called `exsrv+dsk`.

CAUTION

Risk of system degradation

Ensure that you install only the EXPRESS SNA server and the EXPRESS integrated desktop. Installing more packages than `exsrvr+dsk` will cause host communications problems.

Files modified by the EXPRESS installation

The EXPRESS software modifies several files during the installation process. The following table lists the files that EXPRESS accesses during the installation procedure, and describes the modification.

File	Description of the modification
`/etc/profile`	Sets the environmental variables.
`/etc/services`	Sets the TCP port used by EXPRESS.
`/etc/passwd`	Sets up the user `express`.
`/var/EXPRESS/envfile`	Configures kernel-related environmental variables.

Beginning the EXPRESS SNA software installation

The communication software that is loaded is the EXPRESS SNA Server. The EXPRESS SNA Server installation is prompt- and menu-driven.

Removing earlier versions of the EXPRESS software

If the EXPRESS software was previously installed, you must remove this earlier version before installing the new version.

> ***Note:*** You cannot reinstall the software until you remove the earlier version.

To remove the EXPRESS software, follow these steps.

1. Log in to the AP or MRS/AP as **`nortel`**.
2. Click on the UNIX window.
3. Type the command **su,** and press <Enter>.
4. Enter the root password.

   Note: You may also telnet across from the MRS and log in directly as **`root`**.

5. Type the following command:

   `. /etc/profile`

6. Type the following command:

   `express_adm shutdown`

7. Type the following command:

   `pkgrm exsrv+dsk`

8. Remove the directory `/u/exsrv+dsk` and all the files under that directory by issuing the following UNIX commands:
   `cd /u`
   `rm -r exsrv+dsk`

Since the home directory of the user `express` was deleted in step 7, you must create the `/u/express` directory.

Installing the EXPRESS SNA software

Note: If you are upgrading from EXPRESS release 2.5 to release 4.0, back up your EXPRESS config files before you begin. You will be reconfiguring EXPRESS after you install it, and you will find it useful to have your existing files available for reference.

To install the EXPRESS software, follow these steps.

1 Log in to the AP or MRS/AP as **`nortel`**.

2 Click the UNIX window, type the command **`su`**, and press <Enter>.

3 Type the root password.

Note: You may also telnet across from the MRS and log in directly as **`root`**.

4 Issue the UNIX command **`dfspace`** to verify that the system has a minimum of 66.7 Mbytes in the `/u` file system and 50 Mbytes in the `/` file system.

Note: This is a temporary area for installation.

This produces a display similar to the following:

```
/  Disk space: 108.92 MB of 250.20 MB available
(43.43%)

/u Disk space: 645.75 MB of 682.00 MB available
(94.69%)

Total Disk Space: 754.68 MB of 932.20 MB available
(80.96%)
```

5 Insert the EXPRESS CD-ROM into the CD-ROM drive.

6 Create a new user with a user ID of `express`, which will have a home directory of `/u/express`. Use the `scoadmin` utility to complete this process.

a Type **`scoadmin`**, and press <Enter>.

b Type **a** to select `Accounts`.

c Type **u** to select `User`.

d Type **c** to select `Create`.

e In the `User Name` field, type **`express`**.

Note: If the user `express` exists, do not delete (retire) it. When a user is retired, that user name cannot be used again. Ensure that the home directory is set to `/u/express`, then exit and skip to step 7.

f In the `Comments` field, type **`for Host Software`**

g In the `Modify Defaults` field, type **y** to select `Modify Defaults`.

h When the `New Account Parameters` screen appears, press <Enter> until the cursor is on the word `default` of the `Home Directory` field.

i Type **s** to select `Specify`.

j Type **`/u/express`** for the home directory.

k Continue pressing <Enter> until you are asked to confirm the change.

l Type **y** to confirm the change.

m Continue pressing <Enter> until the following appears:

```
Created home directory: /u/express
Created sh file: /u/express/.profile
Greetings mail sent to user: express
Press <Return> to continue:
```

n Continue pressing <Enter> until the following appears:

```
Setting password for user: express
Password change is forced for express.
        Choose password
You can choose whether you pick a password,
or have the system create one for you.
  1. Pick a password
   2. Pronounceable password will be generated
for you
Enter choice (default is 1):
```

o Type **1** to pick a password.

p Type the password **`express`**, and reenter it when you are prompted to do so.

q Press <Enter> when you are prompted to do so.

The user is created.

r Press <Escape> until you return to the Main Menu of `scoadmin`.

s Type **q** to select `Quit`, and answer **y** to confirm.

Note: If the screen appears in reverse video (white characters on a black background), you correct this by entering the `setcolor` command.

7 To back up the UNIX kernel, type **`cp /unix /unix.backup`**

8 To change to `/u` directory, type the following:

```
cd /u
```

9 To copy the image of the EXPRESS software from the CD-ROM to disk, type the following command:

```
# mount /dev/cd0 /mnt
# pkgadd -d /mnt/IU5A -s /u exsrv+dsk
# pkgadd -d /u exsrv+dsk 2>&1 |tee pkgadd.out
```

10 Press <Enter>.

The system displays the following:

```
You are about to begin installation of the EXPRESS
SNA Server. The installation procedure will guide
you through the following steps:

1. Enter information about the environment in
which EXPRESS will be running.

2. Select the protocol drivers which you want to
install.

3. Enter information about the communications
adapter(s) which you will install.

You will then see a summary of all the information
you have provided. Upon your arrival, the actual
installation of the EXPRESS SNA Server will take
place.

Choosing the quit (q) option at any time will abort
the installation procedure.

Choosing the help (?) option will display extra
information.

Press <ENTER> to continue or ‘q’ to quit
installation:
```

11 Press <Enter> to continue.

The system prompts you to enter the type of environment.

12 Press <Enter> to accept the default of `stand-alone`.

The system prompts you to enter the name of the EXPRESS runtime directory for this system.

13 To change the runtime directory, type the following:

`/u/express`

The system prompts you to enter the full path name of the standard backup directory.

14 Press <Enter> to accept the default of `/u/express/db/backups.`

The system prompts you to enter the full path name of the standard backup drive.

15 Type the backup device name:

`/dev/rct0`

The system prompts you to enter the full path name of the standard floppy drive.

16 Press <Enter> to accept the default of `/dev/rdsk/f0q18dt`.

The system displays the following prompt:

```
Do you want EXPRESS to be started at system boot
time?
```

17 Press <Enter> to accept the default of **`y.`**

The system displays the following prompt:

```
Do you wish to add the EXPRESS environment
variables to /etc/profile?
```

18 Press <Enter> to accept the default of **`y.`**

The system displays installation messages.

19 At the next prompt, press <Enter>, and then follow the prompts to continue the installation.

20 When you are asked for information to configure the Arnet card, type the following values:

Interrupt level: **`15`**

I/O address: **`310`**

Memory address: **`DC000`**

21 When your installation is complete, exit from all windows, and then exit from the utility.

22 Power off the system, and then power on.

Allow the system to boot normally.

Starting EXPRESS

Now that you have installed EXPRESS, launch it by following these steps.

1. Log on to the system as **root**.
2. Type the login password.
3. To start the background processes, type the following command:

   . /etc/profile
4. Type the following command at the prompt:

   express_adm start
5. Press <Enter>.
6. To start up the EXPRESS desktop software as a user, type the following command at the prompt:

   express -u express

   Press <Enter>.

   EXPRESS Manager appears. This tool handles configuration functions for the administrator or designated user.

   Note: This starts the GUI EXPRESS desktop software. To start character-based EXPRESS, type the following:
   express -u express -c
   TERM=xterm; export TERM
   For information about navigating through the screens, see the *EXPRESS SNA Server Configuration Guide*.

Note: While EXPRESS is running, you can interact directly with it using the EXPRESS Manager. To exit from the EXPRESS Manager, you must choose `Hide the session` instead of `Exit`. If you use the `Exit` option, it places the running host sessions in the foreground and terminates the session.

Configuring EXPRESS

When you configure EXPRESS, you set up the hardware, communications, and user profiles for the system. This configuration process creates a configuration profile that is stored in the EXPRESS database. EXPRESS software allows the definition and maintenance of multiple configurations; however, only one configuration can be active at a time.

This section provides an overview of the EXPRESS configuration process. For a detailed discussion, refer to the *EXPRESS SNA Server Configuration Guide*.

Defining links, stations, and logical units

Symposium OPEN IVR software supports 254 logical units for each port. To configure a port, you need to define a link, a station, and a pool (or multiple pools) of logical units. For more information about defining links, stations and logical units, refer to the *EXPRESS SNA Server Configuration Guide.*

Establishing a naming convention

Use a naming convention when you are creating links, stations, and logical units during the configuration. The naming scheme requires you to define a prefix that indicates the host port and then add one of the following suffixes to the prefix:

- link
- station
- pool

Here is a sample hierarchy of nodes, links, stations, and LU pools:

```
Node
    sqaLink
    sqa1Station
        sqa1Pool
    sqa2Station
        sqa2Pool
```

The EXPRESS configuration procedure reflects this structure. You configure EXPRESS in this order:

- Define a node.
- Define links.
- Define LUs.
- Define stations.

Refer to the EXPRESS documentation for detailed configuration procedures for nodes, links, LUs, and stations.

Defining links and stations

After you define an SNA node, you can insert a data link under any SNA node in the EXPRESS SNA tree.

> ***Note:*** For more information, refer to the section that explains how to define SDLC in the *EXPRESS SNA Server Configuration Guide.*

The links and stations require different parameters, depending on whether your adaptor is SDLC or Token Ring.

To define links and stations, follow these steps.

1 Define the link using the default configuration parameters, except for the `DSR Wait Timeout` parameter.

2 Decrease the value of the `DSR Wait Timeout` parameter from 600.0 (10 minutes) to 60.0 (1 minute).

3 After you define the link, add a station under the link.

Note: Most of the station parameters depend on values required by the remote host. For SDLC stations, you need to know the secondary station address. For Token Ring, you are asked for a remote MAC address. You determine these parameters at the time of pre-site planning.

4 Set the parameter "`Restart connection if it fails`" to `Yes`.

Note: Setting this parameter to `Yes` means that if the remote system goes down, the station continues to try to connect to it.

Defining LU pools (3270 only)

Each station requires one or more pools of LUs that can be assigned to individual sessions. An LU pool consists of a group of LUs that contain the same parameters. The addresses for the LU pool can be between 2 and 255.

Use "-" to define a range of addresses.

For example: 2-10

Use a comma to list all sets.

For example: 2-10, 12, 15-255

Defining LU 6.2 (5250 only)

When you define an SNA node as an independent LU Type 6.2, the Control Point LU is automatically created. For detailed procedural information, refer to the section that explains how to define the LU 6.2 in the *EXPRESS SNA Server Configuration Guide*.

Defining LU6.2 consists of three tasks:

- defining remote LUs
- defining a mode
- defining the 5250 symbolic destination

Defining remote LUs

The process of defining a remote LU provides EXPRESS with the specific address and characteristics of one host LU.

Define each LU that you want the Terminal Resource Server (TRS) to use. EXPRESS associates three attributes to each LU you define:

- remote LU
- LU pair
- associating mode

You define these attributes in the EXPRESS hierarchical structure.

Remote LU

To define a remote LU, follow these steps.

1 Provide your SNA network ID to EXPRESS.

2 Identify the control point of the host.

3 Identify the address of the LU.

Note: Obtain this information from your source of the host information.

4 Name the remote LU.

Note: This name allows you to refer to the LU by name instead of address. Choose a meaningful name that enables you to make an association with the remote LU. You can include numbers in the names. For example, the LU name Banking LU1 is appropriate if you connect to a banking host application.

LU pair

An LU pair maps the remote LU that you have defined to your Control Point LU.

To map an LU pair, follow these steps.

1 Specify the remote LUs.

2 Specify the local control points.

3 Specify the security characteristics.

4 Name the LU pair.

Note: When naming the LU pair, mention the control point LU. For example, if the Control Point LU is named `IVRComPt`, use this name for the LU pair.

Associating mode

When you associate a mode, you provide specific information about your host connection at the LU level. The most important characteristic of the mode is that it must match the mode that the host expects you to use. EXPRESS offers several modes from which you can choose. You can also define a mode. Refer to the next section, "Defining a mode," for additional information.

Defining a mode

Defining a mode allows you to create an appropriate mode for your host connection if the default modes provided with EXPRESS are inadequate. You obtain this mode name from your source of the host information.

Defining a 5250 symbolic destination

The 5250 symbolic destination ties together remote LU, mode, and control point LU. Use this when you define a session. Refer to the next section, "Defining users and sessions".

Defining users and sessions

The EXPRESS user and session configuration defines the EXPRESS users, and the sessions and supervisory tasks available to these users. Configuration of users and sessions varies with each user population.

For more information, refer to the section that explains how to define user and session profiles in the *EXPRESS SNA Server Configuration Guide*. In addition, the *EXPRESS Integrated Desktop Configuration Guide* provides comprehensive information about defining users for the 3270 and 5250 communication environments.

The Symposium OPEN IVR TRS process handles eight sessions for each EXPRESS user.

Note: The first user needs to be named "`express`" to map to the user created during EXPRESS installations; however, Nortel suggests that you do not assign sessions to the user `express`.

To determine the number of users to define, divide the total sessions required by eight. For both 3270 and 5250 emulation, all sessions are named "a" through "h". For the LU address within 3270 emulation, enter the LU pool name.

Setting limits for terminal emulation

The final step in the EXPRESS configuration process requires setting limits for the terminal emulation service. Refer to the *EXPRESS Integrated Desktop Configuration Guide* for more information about setting limits for this service. You need to set limits for two of the three services listed:

- the standard service
- the service with your system names

For both of these services, set "`maximum users`" and "`EHLLAPI user programs`" to at least the number of users you defined. You can set the limits to more than the number defined, especially if you plan to increase the number later.

Verifying connections to the host computer

Before you bring up the entire host link and run a Symposium OPEN IVR application, test each part of the host link separately. By doing this, you can more easily identify the source of a problem, since you are dealing with a small part of the host link instead of the entire configuration.

This section describes testing to ensure that EXPRESS can connect to and interact with the host.

Checking the status of your links and stations

To bring up the EXPRESS Manager and check the status of your Stations and LUs, follow these steps.

1 Display the EXPRESS Manager by typing the following command:

`express -u express`

Press <Enter>.

The system displays the EXPRESS Manager main window.

2 Click `Control and Administration.`

The system displays the `Control and Administration` window.

3 Click `Control Communications and Diagnostics.`

The system displays the `Links, Stations` window, which displays link and station status.

4 In this window, verify that all links and stations have their status enabled.

5 If a link or station is not enabled, do the following:

a Click the link or station.

b Click the Control menu, and then click `Activate`.

Ensure that the status changes from `Inactive` to `Enabled`. If the status is `Pending`, the system is waiting to contact the host.

Testing 3270/5250 host communications

1 After the EXPRESS software is configured, issue the following commands to restart EXPRESS:

`express_adm stop` (stop the interface)

`express_adm start` (start the interface)

`express_adm check` (check that daemons are running)

`express -u express -c` (start Express)

The system displays the EXPRESS Manager main window.

2 Click `Control and Administration`.

The system displays the `Control and Administration` window.

3 Click `Control Communications and Diagnostics`.

The system displays the `Select Display` screen.

4 Click `Link, Stations` to display the `Link, Stations` menu.

5 Highlight the `Node` line, and then select `Activate` from the `Control` pulldown menu.

Note: To do so, press <F10>, or press <ESC> and then <Enter>, depending on what is displayed for the menu in the `Personal Status` column.

6 Activate the following lines if they do not show `Enabled`:

- `Node`
- `LU - Control Point LU - ONLY FOR 5250 - always configured by Express`
- `Link`
- `Station`

The system displays a confirmation screen.

Note: The `Control Point LU` line is activated only for the 5250. Activate the `Node`, the `Link`, and the `Station`.

7 Press <Enter> with the cursor on `Activate`.

The screen displays `Operation in Progress` and then changes to `Successful completion`.

8 Press <Enter> with the cursor on `Close` to close this window.

9 The `Status` should change to `Enabled` if a successful activation occurs.

10 Continue changing the status on each line that is `Inactive`.

If the `Link` becomes enabled, enable the `Station` as well. When the `Station` is enabled, a successful SNRM or SABME has been passed between the host and the IVR system.

Note 1: If the `Status` does not change to `Enabled`, check the EXPRESS Log. To obtain the EXPRESS Log, choose `Display Log` from the `Control and Administration` menu. The Log provides information about where the problem is occurring.

Note 2: To modify an item in configuration, you must ensure that the item's status is `Deactivated` in the `Control and Communications` screen.

Displaying an interactive session

By initiating an interactive session with the host, you can view the output from the host and verify your host connection.

To initiate an interactive session from the EXPRESS Manager, follow these steps.

1 After all the `Statuses` are `Enabled`, display an interactive session by returning to the `EXPRESS Manager` menu.

2 Select `Interactive Sessions`.

The `Interactive Sessions` menu appears. It lists all the `Sessions` that were configured.

3 Click the session you want to use.

The system displays the sample SNA session window.

- If you are connected to the host, this window displays the main or introductory screen for the host.
- If you are not connected to the host, the window remains empty.

Note: While EXPRESS is running, you can interact directly with it using the EXPRESS Manager. To exit from the EXPRESS Manager, you must choose `Hide the session` instead of `Exit`. If you use the `Exit` option, it places the running host sessions in the foreground and terminates the session.

Starting up EXPRESS daemons and users

Nortel provides a script that you can use to bring up EXPRESS daemons and all defined EXPRESS users for connections with Symposium OPEN IVR TRS. You will find a copy of this template in the following location on the AP or MRS/AP: `/u/nortel/user_data/vtekhost_adm.sna`

Modify the template to match your environment and configuration.Include the desired number of EXPRESS users (one for every eight sessions).

> ***Note:*** The user names in your startup script must match the user names in the TRS session file (`trs.sdf`).

Copy the template to `u/nortel/user_data`, and then make the necessary changes. You install the script in /etc/rc2.d. For more information, see “Editing system startup files” on page 3-30.

```
# Customizable option: location of shell
#
#! /bin/sh
#########################################################
#
#
#
# Customizable option: location of express profile
#
. /etc/profile
#
#error message definitions
#
MESSAGE1=”\nInvalid request usage:vtkhost_adm (start |
stop)”
MESSAGE2=”\nTimeout occurred. Necessary daemons not
active after 5 minutes”
MESSAGE3=”\nAll daemons now running, continuing
processing”
```

```
#

#check to see an action parameter (start or stop) was
#passed in

#

if[ “$#” -lt 1 ]

then

echo$MESSAGE1

exit 1

fi

case “$1” in

‘start’)


#start the express background processes

#


express_adm start


#

#loop to verify all necessary daemons are running then
#start express in hidden mode

#

loopCount=0

retryCount=0

#

#hang out in a loop until all express daemons are

#running. This process will time out after 5 minutes

#
```

```
# Customizable option: Verify all processes below do
# execute on your system
#
#
while [ $loopCount -eq 0 ];do
(ps -ef | grep cdm | grep -v grep)&&
(ps -ef | grep agent | grep -v grep)&&
(ps -ef | grep executive | grep -v grep)&&
(ps -ef | grep watchdog | grep -v grep)

if [ $? -ne 1 ]
then
echo $MESSAGE3
break
else
retry Count=‘expr $retryCount + 1’
if [ $retryCount -eq 12]
then
echo $MESSAGE2
exit 2
else
sleep 5
fi
fi
done
#
# Customizable option: Name of your link.
# Comment out if this is an SDLC connection.
#
```

```
stst<<!
 start datalink sqaLink
!
#
# Customizable option: Name of your station.
#
stst<<!
 start connection sqaStation
!
#
# Customizable option: Name of express users. Add a
# line for each user.
#
  express -u express1 -h
  express -u express2 -h
        ;;
#STOP section of the script. The following logic
#will stop all defined express users, connectivity to
#the host computer and stop all background express
#processes
#

'stop')
  echo "STOP Session Managers"

#
# Customizable option: Name of express users
#
  express -x -u express1
```

```
  express -x -u express2




#
# Customizable option: Name of your Station
#
stst <<!
stop sqaStation
!
#
# Customizable option: Name of your Link. Comment out if
# this is an SDLC connection
#
stst <<!
stop datalink sqaLink
!
express_adm stop
;;
esac
```

Editing system startup files

By editing the system startup files to include the `autostart` option, you ensure that EXPRESS starts when the system is booted and that the EXPRESS users required by Symposium OPEN IVR are enabled.

If you select the `autostart` option when you install the EXPRESS software, the system startup file `S99express` is created in the directory `/etc/rc2.d`. This file contains the following commands:

```
if [$1 start]

then

rm -f/u/express/db/cbmst

fi

su root -c "/u/express/bin/express_adm $1"
```

After you modify the script `vtekhost_adm` to match the EXPRESS configuration, add the following line to the `S99express` file:

```
su root -c "/usr/nortel/3270/vtekhost_adm $1"
```

Note: The user names in your startup script must match the user names in the TRS session file (`trs.sdf`).

Modifying the /etc/rc2.d/S99zgen file

To modify the `/etc/rc2.d/S99zgen` file so that the root user can find `express_adm`, follow these steps.

1. Log on to the system as **`nortel`**.
2. Type the root password.
3. Click on the UNIX window.
4. Type the command **`su`**, and press <Enter>.
5. Unprotect the file by typing the following command:

   `chmod +w /etc/rc2.d/S99zgen`
6. Use an editor to edit the file `/etc/rc2.d/S99zgen`
7. Edit the file by placing the following line before the "`MACHINE . . .`" line:

   `. /u/express/.profile`
8. Save the changes made to the file, and quit the editor.
9. Protect the file by typing the following command:
   `chmod -w /etc/rc2.d/S99zgen`

Chapter 4: Installing and configuring EXPRESS TN3270 or TN5250

This chapter discusses the process of setting up host link communications with EXPRESS TN3270 or TN5250 software. Use this EXPRESS package when you set up TCP/IP connections to the mainframe host and you are using either the TN3270 or TN5250 protocol. This chapter includes information about the following topics:

- system requirements
- site planning
- installing EXPRESS TN3270 or TN5250 software
- setting limits for terminal emulation
- starting EXPRESS TN3270 or TN5250
- configuring EXPRESS TN3270 or TN5250
- editing system startup files
- verifying connections to the host computer

System requirements

Before you install EXPRESS TN3270 or TN5250 software, verify that your application processor is running Symposium OPEN IVR Release 4.0 software. For additional information about host communication support, refer to the *Symposium OPEN IVR System Administration Guide*.

ATTENTION

To use EXPRESS software for host communications, your system must include an Ethernet card.

Site planning

Review your physical configuration requirements and inspect the package contents before you install and configure your system, as described in this chapter.

Verifying required physical configurations

Before you start to install and configure your system, review the required physical configuration of the installation site:

- communications protocol (TN3270 or TN5250)
- number of sessions
- host information (host IP address and, optionally, port number for service)

Verifying package contents

Before you start to install the software, verify that you received the package you ordered, including the following items:

- correct number of logical units
- correct EXPRESS software version
- documentation

Installing EXPRESS TN3270 or TN5250 software

The EXPRESS software package consists of the following components:

- EXPRESS API software development kit
- EXPRESS API runtime
- EXPRESS remote display emulator

Note: For more information about installing the EXPRESS software on your system, refer to the *EXPRESS TN3270/5250-X Administrator's Guide*, *EXPRESS Installation Guide*, and the *EXPRESS Release Notes.*

To install the EXPRESS TN3270 or TN5250 software, follow these steps.

1 Locate or create a temporary location where you can store several megabytes of data.

2 Change the current directory to the temporary directory.

3 Use the `tar` command specified in the *EXPRESS TN3270/5250-X Administrator's Guide* to copy the installation files to the temporary directory.

4 Determine the destination directory for the EXPRESS TN3270 or TN5250 software installation.

Note: This is normally `/u/tnexpress`.

5 Determine the following values:

- total number of concurrent users
- number of EHLLAPI sessions per user
- number of interactive sessions per user

6 Run the install command that is specified in the *EXPRESS TN3270/5250-X Administrator's Guide*. For example, type the following command at the prompt:

`./xtninst -dest pathname -euser user -hprog sess1 -session sess2`

where
`pathname` is the destination path name
`user` is the total number of concurrent users
`sess1` is the number of EHLLAPI sessions
`sess2` is the number of interactive sessions per user

7 Press <Enter>.

8 Install the fonts.

Setting limits for terminal emulation

This section presents information about setting limits for the terminal emulation service. The file `$tnexpress/adm/config/exPSd.cfg` contains definitions for the following values:

- number of users
- number of EHLLAPI user programs
- print support
- host language file
- local language file
- print capability file

Refer to the *EXPRESS TN/3270/5250-X Administrator's Guide* for more information about setting limits for terminal emulation in the file `$tnexpress/adm/config/exPSd.cfg`.

The following example lists the default settings for the configuration file `$tnexpress/adm/config/exPSd.cfg`:

```
NumUsers:          2

NumHUPs:           2

PrintSupport:      True

HostLangFile:      $tnexpress/adm/config/hostlang.cfg

LocalLangFile:     $tnexpress/adm/config/loclang.cfg

PrintCapFile:      $tnexpress/adm/config/printcap.cfg
```

The following example displays the file `$tnexpress/adm/config/exPSd.cfg` for a configuration including 64 simultaneous HLLAPI user programs and 8 EXPRESS users with 8 sessions each:

```
NumUsers:         8

NumHUPs:          64

PrintSupport:     True

HostLangFile:     $tnexpress/adm/config/hostlang.cfg

LocalLangFile:    $tnexpress/adm/config/loclang.cfg

PrintCapFile:     $tnexpress/adm/config/printcap.cfg
```

Starting EXPRESS TN3270 or TN5250

Starting up EXPRESS TN3270 or TN5250 software consists of the following steps:

- starting the EXPRESS PS process
- starting the EXPRESS display emulator

Starting the EXPRESS PS process

Before you start the EXPRESS display emulator, you need to start the EXPRESS TN3270 or TN5250 daemon processes.

1 Log in as **`root`**.

2 Type the following login password:

`root1`

3 To start the EXPRESS daemon processes, type the following command at the prompt:

`expressPS start`

Press <Enter>.

The system displays the following message:

`expressPSd STARTUP COMPLETE`

4 To stop the EXPRESS PS process, type the following command at the prompt:

`expressPS stop`

Note: If you select the `autostart` option when you install Express TN3270 or TN5250, then this process is completed at system boot time.

Refer to the *EXPRESS TN3270/5250-X Administrator's Guide* for more information about starting and stopping the EXPRESS PS process.

Configuring EXPRESS

This section provides an overview of the EXPRESS configuration process. The configuration process involves creating a session file that is used to define all the sessions for a particular user or group. It can be used with the EXPRESS display emulator or by entering the command `start grp`.

Defining users and session groups

The EXPRESS user/session configuration defines the EXPRESS users and the sessions and supervisory tasks available to these users. Configuration of users and sessions varies with each user population.

> ***Note:*** For more information about defining users and session groups, refer to the *EXPRESS TN3270/5250-X Administrator's Guide*.

You define a session file for each EXPRESS user to configure the number of concurrent sessions for the user. When you start a session, a session group name is invoked. This section provides two examples of session files for communication with two different hosts.

Session configuration files

The sample session files `tn3270.cfg`, `tn5250.cfg`, and `vt100.cfg` are released in the `user_data` directory. You copy these files to the 3270 directory and use a text editor (such as `nedit`) to customize these files to define sessions. If the parameters do not change, you can reuse the files `tn3270.cfg`, `tn5250.cfg`, and `vt100.cfg` for each new EXPRESS user.

Sample session file for TN5250 communications

This file is an example of the session file settings for communication with TN5250 host devices. Modify this template to match your environment and configuration. Name this script `express.cfg` and place it in the `/u/nortel/`5250 directory.

```
 # Global parameters for the entire user group#

  accessPermission:   Group        # UNIX permissions for session group
                                   # choices: Owner/Group/All
  numUndefSess:       0            # number of dynamic sessions
                                   supported
  defaultTnHost:      apertus.com  # tn server name or address

# First TN5250 session

   Session:    5250                # TN5250 session
               id:    a            # One character session id
               Name: Session_a     # Name used to identify the session
               iconName: a         # Icon name used for the session
               tnHost:             # telnet server name
               apertus.com
               tnService:7713      # Port number on host used for
                                     telnet service. The default is
                                     telnet. This number can be any
                                     number or name in the file
                                     /etc/services.
```

```
# Second TN5250 session

   Session:    5250                # TN5250 session
               id:    b            # One character session id
               Name: Session_b     # Name used to identify the session
               iconName: b         # Icon name used for the session
               tnHost:             # telnet server name
               apertus.com
               tnService:7713      # Port number on host used for
                                     telnet service. The default is
                                     telnet. This number can be any
                                     number or name in the file
                                     /etc/services.
```

```
# Third TN5250 session

   Session:    5250                # TN5250 session
               id:    c            # One character session id
               Name: Session_c     # Name used to identify the session
               iconName: c         # Icon name used for the session
               tnHost:             # telnet server name
               apertus.com
               tnService:7713      # Port number on host used for
                                     telnet service. The default is
                                     telnet. This number can be any
                                     number or name in the file
                                     /etc/services.

# Fourth TN5250 session

   Session:    5250                # TN5250 session
               id:    d            # One character session id
               Name: Session_d     # Name used to identify the session
               iconName: d         # Icon name used for the session
               tnHost:             # telnet server name
               apertus.com
               tnService:7713      # Port number on host used for
                                     telnet service. The default is
                                     telnet. This number can be any
                                     number or name in the file
                                     /etc/services.
```

Sample session file for TN3270 communications

The file that follows is an example of the session file settings for communication with TN3270 host devices. Modify the template to match your environment and configuration. Name this script `express1.cfg` and place it in the `/u/nortel/3270` directory.

```
# Global parameters for the entire user group#

   accessPermission:  Group       # UNIX permissions for session group
                                  # choices: Owner/Group/All
   numUndefSess:      0           # number of dynamic sessions
                                  supported
   defaultTnHost:     rainbow     # tn server name or address

# First TN3270 session

    Session:    3270               # TN3270 session
                id:    a           # One character session id
                Name: Session_a    # Name used to identify the session
                iconName: a        # Icon name used for the session
                modelNum: 2        # model number for TN3270
                                   # choices 1,2,3,4,5
                tnHost: rainbow    # telnet server name
```

```
# Second TN3270 session

   Session:    3270                # TN3270 session
               id:    b            # One character session id
               Name: Session_b     # Name used to identify the session
               iconName: b         # Icon name used for the session
               modelNum: 2         # model number for TN3270
                                   # choices 1,2,3,4,5
               tnHost: rainbow     # telnet server name

# Third TN3270 session

   Session:    3270                # TN3270 session
               id:    c            # One character session id
               Name: Session_c     # Name used to identify the session
               iconName: c         # Icon name used for the session
               modelNum: 2         # model number for TN3270
                                   # choices 1,2,3,4,5
               tnHost: rainbow     # telnet server name

# Fourth TN3270 session

   Session:    3270                # TN3270 session
               id:    d            # One character session id
               Name: Session_d     # Name used to identify the session
               iconName: d         # Icon name used for the session
               modelNum: 2         # model number for TN3270
                                   # choices 1,2,3,4,5
               tnHost: rainbow     # telnet server name
```

Note: The Symposium OPEN IVR TRS process handles eight sessions for each EXPRESS user. To determine the number of users to define, divide the total number of required sessions by eight. All sessions are named "a" through "h".

Completing your configuration

Before you attempt to bring up the entire host link and run a Symposium OPEN IVR application, test each part of the host link separately. By testing parts of the host link in sequence instead of the entire configuration, you can isolate problems more easily.

Bringing up an interactive session

Bringing up an interactive session with the host is the best way to check your host connection. You see output from the host, and you can actually interact with it.

To bring up the EXPRESS display emulator, follow these steps.

1 Type the following command at the prompt:

`xtnte -sessdef` ***`session_file`***

where *`session_file`* is a preconfigured file that contains user and session definitions for local sessions, TN3270 display sessions, TN5250 display sessions, or printer sessions.

Note 1: In the sample files that begin on pages 4-9 and 4-12, the session_file parameter could be either `express.cfg` for TN5250 or `express1.cfg` for TN3270.

Note 2: "See "Configuring EXPRESS" on page 4-8" section for more information about session files. In addition, refer to the *EXPRESS TN3270/5250-X Administrator's Guide* for more information about starting the display emulator.

2 Press <Enter>.

The Interactive Session window is displayed. This window displays all sessions:

- If you are connected to the host, this window displays the main or introductory screen for the host.
- If you are not connected to the host, the window remains empty.

Bringing up an interactive session for a particular user

To bring up an interactive session for a particular user, follow these steps.

1 Type the following command:

`xtnte -group` ***`username`*** **`-sessdef`** ***`session_file`***

where

`username` is the name of a user defined in the `vtekhost_adm.tn` file

`session_file` is the name of the session file appropriate for the session type

Note: This command displays a menu listing all the sessions available. Selecting a session displays an interactive screen for that session. You can use this screen

- to verity the host connection
- to cycle through the host application
- to watch the interaction between the AP software and the session

CAUTION

Risk of data loss

If you run this command when the AP software is active, do not quit `xtnte`. If you quit `xtnte`, the system terminates the connection between the client and the server.

2 To select another session, select Open from the File menu.

The system displays the session list again.

3 To remove the window from your screen, click the Minimize button using MB1.

Note: The minimize button is a small square located in the upper right corner of the window. Clicking this button closes the window. Its icon then appears in the Icons window.

4 To restore a window from its icon, double-click the icon using MB1.

Editing system startup files

If the `autostart` option is selected when EXPRESS software is installed, the system startup file `S99expressPS` is created in the directory `/etc/rc2.d`. This file contains the following commands:

```
su root -c "/u/tnexpress/bin/expressPS $1"
```

After you modify the script `vtekhost_adm.tn` to match the EXPRESS configuration, add the following line to the file `S99expressPS`:

```
su root -c "/u/nortel/3270/vtekhost_adm.tn $1"
```

Starting user groups in the background

In order for the Symposium OPEN IVR software to use the EXPRESS TN3270/TN5250 session for connection to a host computer, you start user groups in the background by using the script `vtekhost_adm.tn`. You will find a copy of this template in the following location on the AP or MRS/AP: `/u/nortel/user_data/vtekhost_adm.tn`

This script verifies that the EXPRESS daemons are running and then starts the necessary user groups. Before using this script at your site, modify the script `vtekhost_adm.tn` to reflect your needs. Include the following:

- the correct number of user groups
- the appropriate session files

For use with the Symposium OPEN IVR TRS, each user group can support up to eight sessions. If all sessions contain the same parameters, you can use the same session file with each user group.

Nortel provides a template for a script that automates the EXPRESS startup background processes `PSd` and `PRs` and enables the links and stations of a particular configuration. This section includes an example of the template file `vtekhost_adm.tn`

Modify this script to match your configuration. Copy the template to the `/u/nortel/3270/vtekhost_adm.tn directory`, and then make the necessary changes. Modify this file to include the correct number of EXPRESS users for your particular configuration.

```
# Customizable option : location of shell
#
# ! /bin/sh
############################################################
#
#
#
# Customizable option : location of express profile
#
. /etc/profile
#
# error message definitions
#
MESSAGE1=”\nInvalid request usage: vtekhost_adm.tn
(start|stop)”
MESSAGE2=”\Timeout occurred.Necessary daemons not
active after 5 minutes”
MESSAGE3”\nAll daemons now running, continuing
processing”
#
# check to see an action parameter (start or stop) was
passed in
#
if [ “$#” -lt 1 ]
then
echo $MESSAGE1
exit 1
fi
case “$1” in
```

```
‘start’ )


#
#start the tn5250 background process
#
expressPS start
#
#Make sure the tn5250 background process is running
#
loopCount=0
retryCount=0
#
#hang out in a loop until all express daemons are running.
This #process will timeout after 5 minutes
#
# Customizable option : Verify all processes below do
# execute on your system
#
while [ $loopcount -eq 0 ] ; do
(ps -ef | grep expressPSd | grep -v grep) &&
(ps -ef | grep expressPRd | grep -v grep)


if [ $? -ne 1 ]
then
echo $MESSAGE3
break
else
retryCount=‘expr $retryCount + 1‘
if [ $retryCount -eq 12 ]
then
```

```
echo $MESSAGE2
exit 2
else
sleep 5
fi
fi
done
#
# Customizable option : Name of express users
#
echo “START session Managers”
startgrp -name express10 -cfg express.cfg
echo “User express10 started”
startgrp -name express20 -cfg express.cfg
echo “User express20 started”
;;

#STOP section of the script. The following logic
#will stop all defined express users, connectivity to
#the host computer and stop all background express
#processes
#
‘stop’ )
echo “STOP Session Managers”



#
# Customizable option : Name of express users
#
```

```
stopgrp -name express10
stopgrp -name express20
#
#Stop background TN5250 processes
#

expressPS stop

;;
esac
```

Chapter 5: Configuring Symposium OPEN IVR host communications

Two configuration files define the EXPRESS configuration to TRS and map EXPRESS LUs to application programs and templates. These files—`trs.conf` and `trs.sdf` —integrate the EXPRESS software with the Symposium OPEN IVR TRS process.

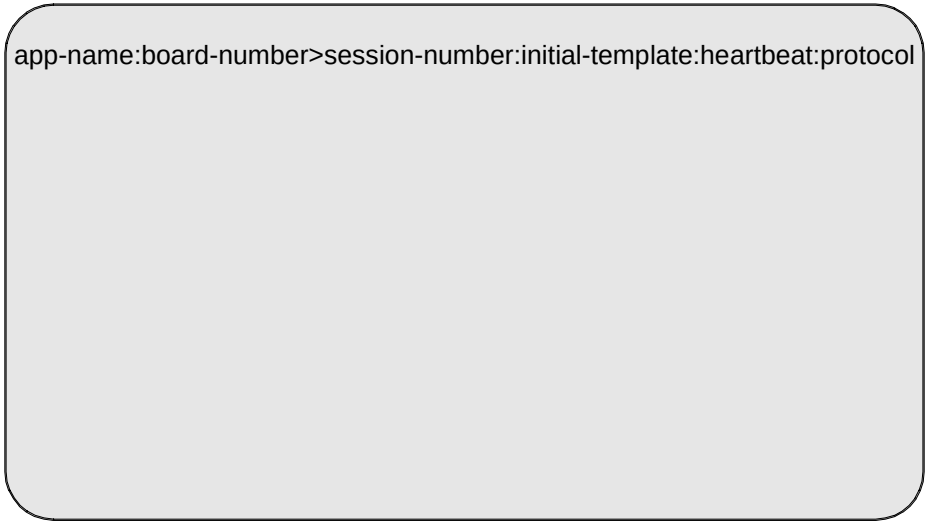
This section discusses the following topics:

- Setting up the trs.conf file.
- Setting up the trs.sdf file.
- Specifying separate IDs and passwords for each LU.
- Bringing up Symposium OPEN IVR, including TRS.
- Mapping channels to host sessions.

Setting up the trs.conf file

The `trs.conf` file specifies the Initial-Action template for each session you define on the Symposium OPEN IVR application processor. The `trs.conf` file must reside in the `/u/nortel/3270` or `/u/nortel/`5250 directory. The following figure shows the syntax for the `trs.conf` file.

Figure 5-1: trs.conf file syntax



Notes on the file syntax

Here are some guidelines for the file syntax:

- Place the colons (:) and the greater than (>) symbols, which signify field delimiters, in the specified positions without any additional white space.
- The entry *`app-name`* identifies the mainframe application to which you are assigning sessions. Enter the same name you enter in the Action templates that are executed by the Symposium OPEN IVR application.
- Enter 0 for the first (or only) *`board-number`* entry, and 1 for the second.

- The *session-number* entry determines the addresses of LUs to activate at start-up. Enter the number as a range of addresses from 2 to 255 (for example, 2-33).
- The *initial-template* entry identifies an Initial-Action template (without the .act file extension) for setting the start-up action for the specified sessions when connecting to the host computer. The start-up action brings the specified sessions to the screen on the host computer where the Action templates start.
- You can specify an optional heartbeat action for the application specified by the *app-name* entry. You can use this feature to send an indication to the host that a session is still active. Some hosts log out sessions that remain idle for a period of time. You can also use the heartbeat action to check connectivity, verifying that the session remains on the appropriate screen. Typically, the heartbeat action contains a Logout-Action template which brings the session back to the login screen if the connectivity with the mainframe fails. If you do not need a heartbeat action, enter a hyphen (-) for this action.
- The protocol entry indicates the communications protocol being used by *app-name*. Set this required field to 3270 or 5250.

Sample trs.conf file

In the following example, the Initial-Action template files login.act and signin.act are defined and reside in the /u/nortel/3270 or /u/nortel/5250 directory. An application developer may want to access the following four applications on the host computer:

- accounting, to access the accounting software
- market, to track stock market activity
- banking, to retrieve credit balances
- airline, to purchase tickets on a local commuter carrier

The application names shown here are not necessarily the actual names assigned on the host computer, but they are the names assigned on the application processor in the trs.conf file and in all Action templates that use these applications.

In this example, you set up trs.conf to initialize the sessions as follows:

- Initialize sessions 2-3 for `accounting` with `signin.act`.
- Initialize sessions 4-8 for `market` with `login.act`.
- Initialize sessions 9-10 for `banking` with `login.act`.
- Initialize sessions 15-17 for `airline` with `signin.act`.

The following figure illustrates a specific `trs.conf` file.

Figure 5-2: Sample trs.conf file

```
#trs.conf file to set up sessions for the application processor
accounting:0>2-3:signin:ping@60:5250
market:0>4-8:login:-:5250
banking:0>9-10:login:-:5250
airline:0>15-17:signin:ping@60:5250
```

Setting up the trs.sdf file

TRS requires the `trs.sdf` file when using the EXPRESS host connectivity package because the file `trs.sdf` maps the short name of each session to the LU number.

> ***Note:*** Refer to the documentation accompanying the EXPRESS package for information on configuring EXPRESS for your environment.

Configuring EXPRESS provides you with the values for the `trs.sdf` file. The following figure displays the format for the `trs.sdf` file.

Figure 5-3: tsr.sdf file format

```
U:Username

Session Short Name    LU Number

Session Short Name    LU Number

Session Short Name    LU Number

•                     •

•                     •

•                     •
```

You can list up to eight sessions and LUs under each EXPRESS user name.

Specifying separate IDs and passwords for each LU

If your host application requires the use of separate login IDs and passwords, list as many different logins and passwords as you plan to use in the `lubuf.dat` file.

If your host application only needs a single login ID and password for each session, place the login ID and password in a screen template. In the single ID case, you do not need to create the `lubuf.dat` file; use the screen template directly. The `lubuf.dat` file must reside in the directory `/u/nortel/3270` or `/u/nortel/`5250 directory. The file may contain up to four variables for each individual session, two of which are `LOGIN_ID` and `PASSWORD`. If you require fewer than four variables, leave the unused variables blank.

The field definitions follow for the `lubuf.dat` file:

```
board# session# &LOGIN_ID &PASSWORD &LU_BUF1 &LU_BUF2
```

Separate each field by space(s), or press the <Tab> key. Each line of the file specifies the contents of defined variables for each particular session number, as displayed in the following example:

```
screen_name 1,2 validation_tag

2,20          -         &LOGIN_ID

3,20          -         &PASSWORD
```

If the screen template appears as in this example, then the login ID is located in the `lubuf.dat` file and copied to the screen location `2,20`. Likewise, the password is located in the `lubuf.dat` file and copied to the screen location 3,20. Using this example, the contents of the `lubuf.dat` file appear as follows:

```
0   2   vtek001 voice1

0   3   vtek002 voice2
```

The first line of the file indicates that the contents of the `&LOGIN_ID` variable and `&PASSWORD` for `board 0`, `session 2` are `vtek001` and `voice1`. In this example, only two variables are required; therefore, you do not need to enter values for the optional variables `&LU_BUF1` and `&LU_BUF2`.

Bringing up Symposium OPEN IVR, including TRS

This section includes information about the following topics:

- using start-up switches
- testing communication with Symposium OPEN IVR

Using start-up switches

To modify specific operations, TRS uses the switches listed in the following table.

Switch	Description
q	Message ID used by the process. For Symposium OPEN IVR Version 2.00 or greater, this value is 98; otherwise, it is 97. This switch is required.
v	Verbose flag. Causes output to be generated to `trs.log`.
r	Revision flag. Displays the current TRS version.
c	Check flag. Displays information on the TRS version and verifies the accuracy of the application templates and action files. Exits after completing the check.
b	Background mode. Application default.
e	Envoy mode.
t	Ignores control characters when filling output buffers.

Completing the Symposium OPEN IVR host communication configuration

Use the procedure in this section to ensure that Symposium OPEN IVR software interacts correctly with EXPRESS software.

TRS start-up file

When Symposium OPEN IVR is started or reset, the TRS process writes initialization results to the file `trs.start-up` in the `/u/nortel/3270` or `/u/nortel/5250` directory. When the initialization process is successful, the system displays the number of sessions that are started. Make sure that this number matches the number of processes listed in the `trs.conf` file. Start TRS in verbose mode if the number of processes does not match the number of processes listed in the `trs.conf` file.

The procedure requires instructing the host link to run in verbose mode. In addition to confirming that the host link and Symposium OPEN IVR software can communicate, the process provides significant data about how the host link is performing. This information assists in isolating host link problems.

To start `trs` in verbose mode, follow these steps.

1. Change the current directory to the Symposium OPEN IVR home directory by typing the following command at the prompt:

   `cd /u/nortel`

   Press <Enter>.

2. Type the following command to change the line in the start-up file:

   `./trs -b -v`

3. Press <Enter>.

4. Type the following command to tail the `trs.log` file:

   `tail -f trs.log`

   Press <Enter>.

   Note: Use the <Delete> key to stop watching the `trs.log` file.

5. Start the host link, if it is not already started.

6 Reset the Symposium OPEN IVR software.

7 Load and run an application that utilizes host connectivity.

Note: The entries in the `trs.log` file indicate communication between the Symposium OPEN IVR software and the host link. The information in the `trs.log` file provides information for debugging and troubleshooting.

Mapping channels to host sessions

The optional file `$VOICE_HOME/3270/map.dat` enables installations to associate channels with specific host sessions. When a host transaction is initiated, the host communications module determines whether the channel on which the call arrived is mapped to a specific session.

If the channel is mapped, then the transaction proceeds over the mapped session. If the channel is not mapped, then session selection is determined by the content in the `trs.conf` file. The syntax in the `map.dat` file is as follows:

`channel_number session_number`

Mapping channels to session is useful during application development. The capability enables application developers to configure an association between the channel of the incoming call and the session selected to communicate with the remote host. However, if the mapped host session is not available, any call arriving over the channel is unable to initiate host transactions.

If your EXPRESS configuration uses LU pools, ensure the sessions defined in the `map.dat` file are not in the LU pool, as the channel may not be mapped to the same host LU when the AP software is restarted.

Sample configuration and template files

You will find sample TRS configuration files and templates in the user_data directory. You can modify these files using the `vi` editing tool.

Note: Move these files to the 3270 directory before modifying them.

File name	Description
`trs.conf`	Defines which LUs are used with each application.
`trs.sdf`	Defines user names and LUs for each user. These names must match the names used in `vtekhost_adm`.
`*.act`	Sample action template.
`*.scn`	Sample screen template.

Chapter 6: Creating template files

The Terminal Resource Server (TRS) uses Action and Screen templates to maneuver through the screens of a host application. These templates provide the host with the same input that would be entered by a terminal operator. The templates also receive the output from the host.

In the application development process, you create these templates. Before creating the templates, you need to understand the transactions that occur in the host application.

This chapter provides information about the following topics:

- determining the actions a terminal operator performs to enter and retrieve information
- creating the different Action templates that define the sequences of host application screens for each transaction
- creating the Screen template files that define the sequence of fields encountered on each screen

Overview of Action templates

Imagine that you are an operator sitting at a terminal. In order to perform a specific task, you type information and press function keys until the desired task is complete. Perhaps a caller asks you to look up a customer's account balance or enter an order. The steps you perform at the terminal enable the application on the host computer to complete the transaction.

To perform a transaction, the operator first logs in to the system from a terminal and then types information into specific fields at specific locations on a screen. Depending on the application, the operator can perform many transactions before logging out.

To develop a voice application that accesses the same information as a terminal operator, you need to tell Symposium OPEN IVR software how to execute the same series of actions that the terminal operator executes. You provide this information in ASCII text files called Screen and Action templates.

Purpose of Screen and Action templates

Screen and Action template files provide the layout and content of each screen in the host application as the terminal operator sees it. The figure on page 6-3 compares a transaction performed by a terminal operator to one performed by a customer calling in to a voice-response system.

Figure 6-1: Sample transaction — terminal operator versus a voice response system

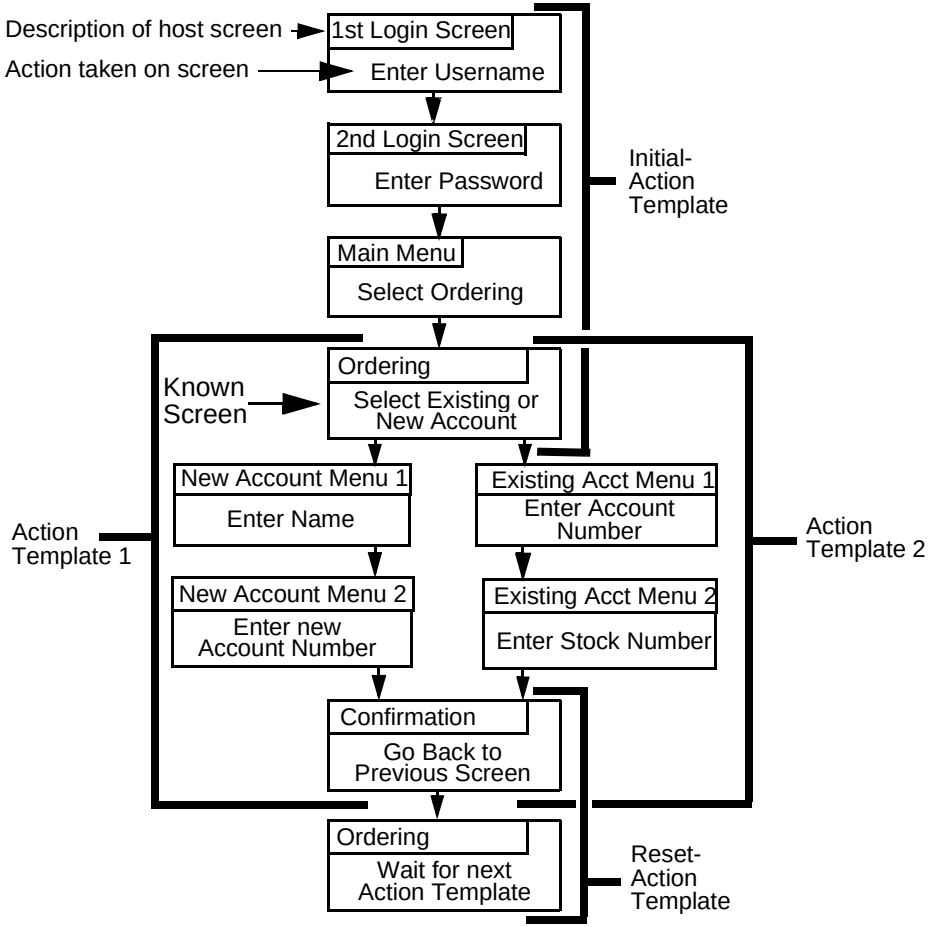
An operator follows this sequence to retrieve data:	A customer follows this sequence to retrieve data:
1 Starts the accounting application.	1 Calls into the MRS, and activates a voice application.
2 Selects the Accounts Receivable menu option.	2 When prompted, selects the Accounts Receivable option from a menu prompt.
3 Asks the caller for account information.	3 When prompted, enters account information. At this point, the application sends a request for information to the TRS. The TRS then executes the Action template for this specific transaction.
4 Enters the caller's account number.	4 Hears the playback of requested data information.
5 Reports the balance when the customer's information appears on the screen.	5 Hangs up.

The host link uses Action templates to navigate through the different screens of a host application.

Action templates describe the sequence of screens traversed to perform a specific transaction. Screen templates validate each screen, define the fields on the screen that require data, and define all keystrokes required for the screen.

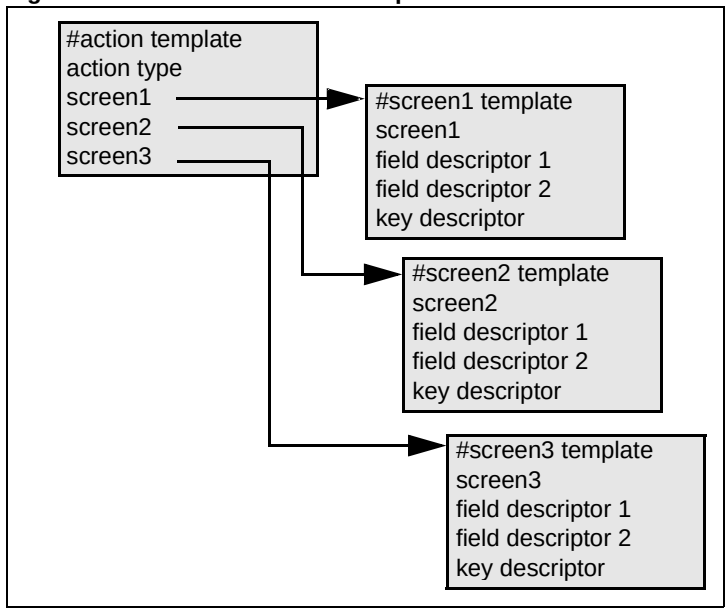
The following figure demonstrates how Action templates work.

Figure 6-2: How Action templates control different segments of a transaction



The following figure shows how Screen templates relate to Action templates.

Figure 6-3: Action and Screen templates



Purpose and structure of the trs.conf file

You create the `trs.conf` file to enable the TRS to determine the identity of each screen, as well as to handle inconsistencies within screens.

The Action templates, Screen templates, and `trs.conf` file are ASCII text files that use a simple syntax to define the screen flow and input/output fields. The sections that follow provide a detailed explanation of the templates, as well as the information necessary to create the `trs.conf` file.

For more information about the `TRS file`, see pages 5-2 to 5-5.

Note: If you back up your template files, do not store the backup files in the `/u/nortel/3270` or `/u/nortel/5250` directory or in any subdirectory under one of these directories. The TRS searches the `/u/nortel/3270` or `/u/nortel/5250` directory and any subdirectories for files with the `.act` or `.scn` extensions.

SNA-based applications often contain format inconsistencies that make it difficult for the TRS to efficiently determine when the host application is ready for input and what region of the host screen contains vital information. These inconsistencies are the result of the character-based format of the SNA protocols. SNA communication protocols cannot notify the TRS that host-data transmission ended. However, the terminal operator knows when the host transmission ends because the requested information appears on the screen. From the TRS perspective, nothing in the SNA protocols provides notification that host output is completed. When the TRS encounters a stream of data from the host, it uses this stream to determine the identity of each screen, in addition to locating the end of each screen.

Types of Action templates

This section provides information about the different types of Action templates:

- Initial-Action templates
- Action templates
- Reset-Action templates
- Logout-Action templates
- Keep-Alive or Heartbeat template

Each type of Action template is suited to a specific portion of a transaction. Each template can handle several host screens.

Initial-Action templates

The host link executes the Initial-Action template automatically when Symposium OPEN IVR starts up. This template handles login and navigates to the screen from which all regular Action Templates start. For more information, see “Initial-Action templates” on page 6-16.

Action templates

The regular Action templates start where the Initial-Action template leaves off. Create one Action template for each transaction. For more information, see “Action templates” on page 6-17.

Reset-Action templates

The Reset-Action template moves a completed transaction back to the screen from which an Action template can perform another transaction. For more information, see “Reset-Action templates” on page 6-18.

Logout-Action templates

The Logout-Action template can recognize any screen in the host application and from that screen can move the transaction back to the first login screen. Then the Initial-Action template can take control.

All Action templates operate on the principle that the host screens appear in a consistent sequence, and that each screen is formatted consistently. If the host application presents screens in an erratic manner or if screens are not formatted consistently, you may have difficulties executing your templates. If this occurs, your Action or Reset-Action templates can stop on an unexpected screen or the LU can enter a locked state. For these reasons, create a Logout-Action template. For more information, see “Logout-Action template” on page 6-20.

Keep-Alive or Heartbeat templates

You can specify an optional Keep-Alive or Heartbeat action for a transaction. You can use this feature to send an indication to the host that a session is still active, since some hosts log out sessions that remain idle for a period of time. For more information, see “Keep-Alive or Heartbeat Action template” on page 6-23.

Creating Action templates

The first step in creating Action Templates is to gain an understanding of the host application. To do this, you need to access the host application through a terminal or an EXPRESS interactive session.

1 Navigate through the host application the same way you want your Symposium OPEN IVR application to navigate. You may want to consider taking notes or creating a flowchart of the screens you encounter. A flowchart of the host application provides a framework for creating Action and Screen templates.

2 Analyze the transactions as you examine the host application. In other words, note the screens you encounter when performing a specific transaction from start to finish.

Note: The first step performed by the terminal operator is not performed by the Action template. It is performed by the Initial-Action template, described later in this chapter. The Initial-Action template handles the login and moves the application to the appropriate screen to begin the transaction.

3 You may want to designate one of the screens of the host application as a “Home” or “Known” screen. The known screen occupies an essential role in the creation of some of the specialized Action templates. Typically, the Initial-Action template logs in the user and displays the known screen.

Action template guidelines

Action templates are ASCII text files created by using a text editor. When you are creating an Action template, ensure that you follow these guidelines:

- Action templates reside in the `/u/nortel/3270 or /u/nortel/5250` directory, depending on your software.
- Assign the extension `.act` to the file name. For example, if you are using 3270 software, the path name for an Action template called `getbalance.act` is `/u/nortel/3270/getbalance.act`

Note: If you back up your template files, do not store the backup files in the `/u/nortel/3270` or `/u/nortel/5250` directory, or in any subdirectory under one of these directories. The TRS searches the `/u/nortel/3270` or `/u/nortel/5250` directory and any subdirectories for files with the `.act` or `.scn` extensions.

Sample host application

You will find an example of a host application on the next few pages. Although based on a simple host application, this example demonstrates the principles you need to understand to effectively create Action templates. Figure 6-4, “Placing an order using an existing account”, shows the host application performing a transaction in which a caller places an order. Figure 6-5, “Creating a new account” shows the host application creating a new account for a caller.

Figure 6-4: Placing an order using an existing account

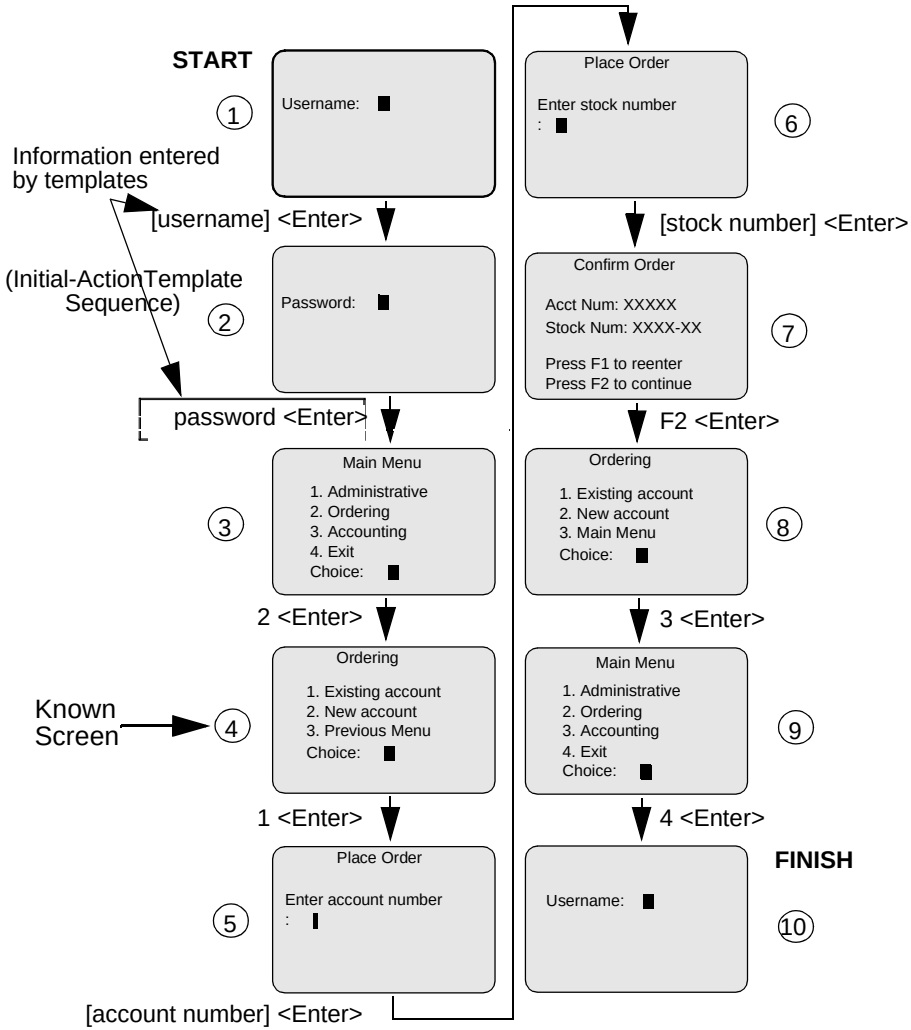
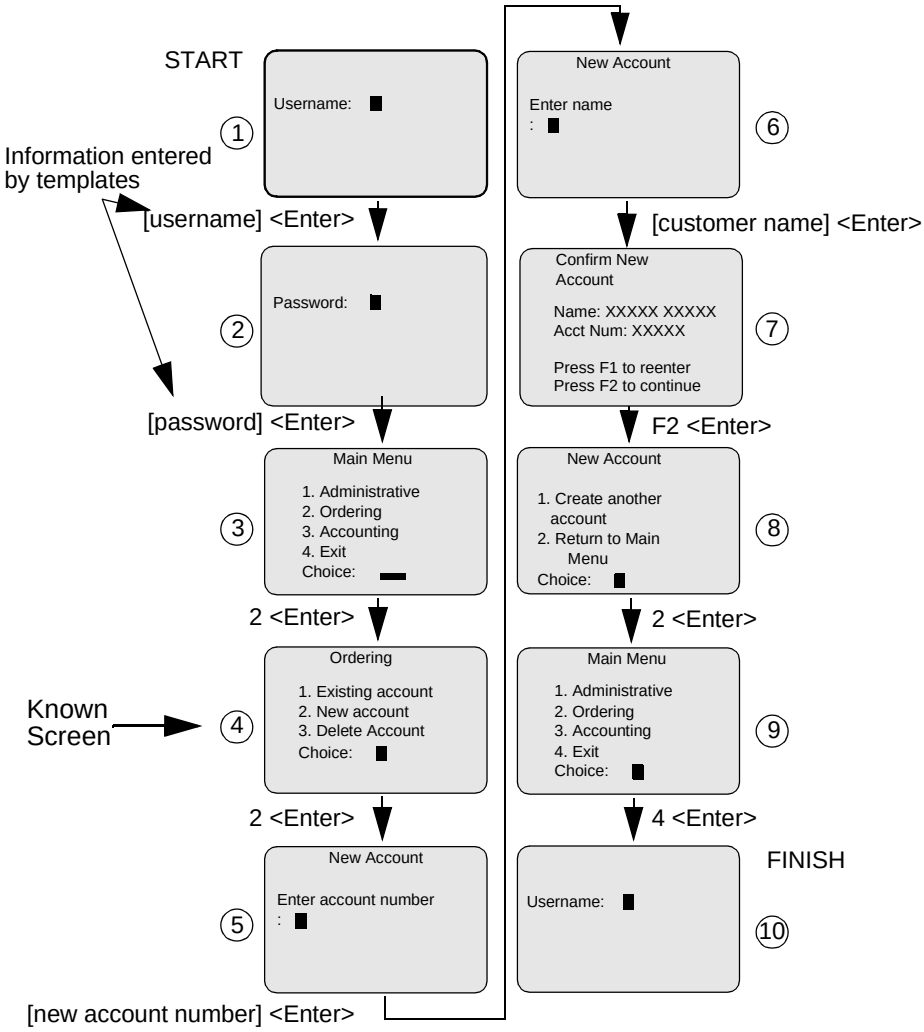


Figure 6-5: Creating a new account



Note that the transactions shown in Figure 6-4, “Placing an order using an existing account”, and Figure 6-5, “Creating a new account**”,** share certain screens. The first four screens are the same for both transactions. Therefore, a single Action template, referred to as the Initial-Action template, handles the beginning of the transaction and navigates through these common screens.

In this example, the Initial-Action template moves the transaction up to the “Ordering” screen (screen 4). This Action template parks the transaction at screen 4, where different transactions start.

At screen 4, the transaction can branch to an “Existing account” transaction or to a “New account” transaction. Thus, a different Action template takes over at this point, depending on the type of transaction your application is trying to complete. For this example, you create one Action template for the ordering transaction, and one Action template for the new account transaction. Note that these Action templates pick up on the screen where the previous template, the Initial-Action template, leaves off.

The transactions once again merge after screen 8. At this point, another Action template can take over and move the transaction back to the Main Menu (screen 9) so that another transaction can begin. Since all Action templates need to start from a specific screen (screen 9 or screen 3 in this example), you define an Action template that moves the transaction back to the screen from which it started. This Action template, called the Reset-Action template, handles resetting the transaction so that another transaction can begin.

Action template syntax

All Action templates follow the same syntax rules. The following displays the syntax of an Action template:

```
#comment
action_name app_name reset_action logout_action
manual
screen template
screen template
screen template
```

where *action_name* is the Action name, *app_name* is the Application name, *reset_action* is the Reset Action, *logout_action* is the Logout Action, and *manual* is the Manual mode.

Understanding the syntax

This section provides information about the syntax of an action template:

- #comment
- action_name
- app_name
- reset-action
- logout-action
- manual
- screen template

#comment

The first line of the Action template is a comment. The comment line is not required but is recommended to describe the purpose of the Action template. It is good practice to use comment lines so that you or others can easily find and make changes to a template in the future. Comment lines start with the # symbol and can be embedded anywhere in the Action template.

action_name

The action name is the Action template file name without the `.act` extension. The action name is required to begin the transaction. For example, if the file name of the Action template is `getbalance.act`, then you type **`getbalance`** for the action name.

app_name

When you install the host link on your application server, you create a `trs.conf` file that assigns TRS session numbers to the application on the host computer.

Choose a name for the host computer application. It does not need to match any actual application name on the host computer, but you must use the name consistently throughout your Action templates.

As an example, this guide uses the application name `ord.vpf` to represent an Ordering application for the host computer.

The application name that you specify in the Action template must exist in the `trs.conf` file. Symposium OPEN IVR software passes the application name to the host link, which uses the name to start the appropriate session with the host computer.

reset_action

In this field, type the name of the Reset-Action template that moves the transaction back to the point from which the Action template started. Again, enter the name without the `.act` extension. For example, if the file name of your Reset Action template is `reset.act`, you type **`reset`** in this field.

The Reset Action template is executed upon successful completion of the Action template.

logout_action

In this field, type the name of the Logout-Action template that moves the transaction back to the point where the Initial-Action template can execute. Again, type the name without the `.act` extension. For example, if the file name of your Logout Action template is `logout.act`, you type **`logout`** in this field.

The Logout-Action template is executed if the Action or Reset-Action templates fails, or if the Action template fails and there is no Reset-Action template specified. If the transaction succeeds, the Logout-Action template is not executed. After the host link uses the Logout-Action template to return a failed transaction to the initial screen, it executes the Initial-Action template after 30 seconds.

manual (optional mode)

The optional manual mode entry allows you to attach a session to a particular channel. You can then use the same session for consecutive Symposium OPEN IVR transactions. This type of session is not released when the transaction is finished. To exit from manual mode, execute a COMA cell in your Symposium OPEN IVR application, or process another Action template that does not contain the manual mode option. Chapter 7, "Using the COM cells to access the host computer", describes how to use the COMA cell.

To specify manual mode, type **`manual`** after the Logout-Action template field. If you omit `manual`, automatic mode is used for the template. In automatic mode, the session assigned to the Action template is free for use when the transaction is completed.

> ***Note:*** Do not specify a Reset-Action template in an Action template that uses manual mode. Manual mode is designed to stay at a specific screen. The next transaction received by the session starts at the last screen of the Action template that used manual mode. This next transaction should use automatic mode and specify a Reset-Action template that brings the session back to the starting screen of the first Action template that specified manual mode.

screen template

In these fields, you type the names of the Screen templates following these guidelines:

- Enter the name of the Screen template without the file name extension.
- Enter the Screen templates in the exact order in which they appear during the transaction.
- List each Screen template on a separate line.

Initial-Action templates

The Initial-Action template serves two purposes:

- It simplifies the development of host link applications by setting a starting point for the Action templates.
- It allows your applications to process host-link requests more quickly, since the transaction is waiting at a screen nearer the information screen. Therefore, the host link does not need to navigate through as many screens to retrieve information.

If your transactions do not start from the same screen, you can create multiple Initial-Action templates. Multiple Initial-Action templates can address a variety of different situations. You may have a situation where all your transactions except one begin at the seventh screen of your host application. This one transaction begins at the second screen. Thus, instead of creating one Initial-Action template that sets the starting point at screen 2 and forces all your transactions (except for one) to move through five additional preamble screens, you can create two Initial-Action templates. The first of these Initial-Action templates puts a certain number of terminal sessions at screen 2, while the second sets a certain number at screen 7. This places all transactions at the optimum starting point.

As a further example, if half of your transactions begin at screen 8, a quarter of them begin at screen 6, and a quarter begin at screen 2, then you define three Initial-Action templates. Each Initial-Action template places a portion of the terminal sessions at the appropriate screen.

You assign Initial-Action templates to terminal sessions in an ASCII text file named `trs.conf`. This file automatically executes the appropriate Initial Action templates at Symposium OPEN IVR startup.

> ***Note:*** If you are creating applications for the TN3270, include transactions in your Initial-Action template that enable your system to recover successfully if your Telnet server goes down.

Sample Initial-Action template

The Initial-Action template follows the same format and syntax as defined in the "Action template guidelines" on page 6-8. This Initial-Action template works with the sample Action template in Figure 6-7 on page 6-18.

Figure 6-6: Sample Initial-Action template

```
# Sample Initial-Action template. Filename:login.act
#
# action  app   reset      logout    manual
# name  name  action      action     mode
#-------------------------------------------------------
 Login   Ord   GoToLogin  FindLogin

# Screen
# Templates
#-------------------------------------------------
 EnterUName
 EnterPswd
 SelOrder
```

Action templates

Action templates are designed to start at the point where the Initial-Action template leaves off. You create a separate Action template for each unique transaction within the host application. The Symposium OPEN IVR application invokes Action templates, so you need a one-to-one correspondence between your Action templates and the different transactions your application tries to accomplish.

An Action template specifies the same sequence of screens that you see as a terminal operator. Whenever the host link references an Action template, it executes the Screen templates listed in the Action template, moving through the host application just as a terminal operator navigates through screens to handle a transaction.

The following example illustrates an Action template file for a transaction for placing an order.

Figure 6-7: Sample Action template

```
# Sample Action template. Filename:PlaceOrd.act
#
# action   app   reset     logout    manual
# name   name  action     action    mode
#-------------------------------------------------------
 PlaceOrd  Ord   GoToMenu FindLogin

# Screen
# Templates
#-----------------------------------------------
 EnterName
 EnterAcct
 Confirm
```

Reset-Action templates

Since Action templates start where Initial-Action templates leave off, there must be a mechanism to return the transaction to this starting point after the Action template finishes. This mechanism is the Reset-Action template.

The Reset-Action template returns a transaction to the screen where the Initial-Action template left off and the Action template started.

The Reset-Action template can return the transaction to the Home Screen in a variety of ways, depending on how the host application is designed. The Reset-Action template may enable the user to return to the “Home” screen simply by pressing a single key, or it may step the user back through all the screens traversed by the Action template. The Reset-Action template can also combine these actions. When you are planning your template, choosing the shortest consistent route to the “Home” screen is the best strategy.

If you want to eliminate a Reset-Action template, insert a hyphen (-) in the Reset-Action entry in your Action template. If the transaction succeeds and a Reset-Action template is not specified, then the host computer application remains parked at the screen where the transaction ends. If you do not specify a Reset-Action template and the transaction fails, the Logout-Action template is executed. The Logout-Action template is described later in this chapter.

When you create a Reset-Action template, do not specify Reset-Action or Logout-Action templates within it.

The following figure illustrates the Reset-Action template that is specified in the sample Action template in Figure 6-7.

Figure 6-8: Sample Reset-Action template

```
# Sample Reset-Action template. Filename:GoToMenu.act
#
# action  app  reset  logout  manual
# name   name  action  action   mode
#-------------------------------------------------
 GoToMenu  Ord   -    -

# Screen
# Templates
#-------------------------------------------------
 ExitConfirm
 ExitAcct
```

Logout-Action template

The Logout-Action template executes only if one of the Action templates fails, which occurs if one of the Screen templates specified in the Action templates did not navigate through its screen.

The Logout-Action template moves the transaction from an unknown state to a known state. The Logout-Action template determines the progress of the transactions. When the Logout-Action template is invoked, it begins executing a series of Screen templates. When one of the Screen templates "hits" (meaning it affected the host screen), the sequence of the Screen templates is such that it begins moving the transaction back to a logout. However, when host errors occur, the sequence of the host screens may be incorrect. There is no guarantee that a "hit" on one screen automatically puts a transaction on the path to a logout.

The Logout-Action template executes a sequence of Screen templates. This sequence of Screen templates is executed in order, regardless of the success or failure of each Screen template. If the final Screen template in the sequence is successfully executed, the Logout-Action template considers the logout complete and invokes the Initial-Action template after 30 seconds. If the final Screen template in the sequence is not successfully executed, the sequence is repeated.

Create a successful Logout-Action template by placing Screen templates in a sequence that enables the best movement for the transaction to a logout. For example, ensure that every host screen contains a certain key that always exits or places you on a known screen. Then your Logout-Action template specifies only a single Screen template, one that presses that key. But not every screen contains an "Exit" key. Perhaps half the host screens contain "Exit" keys and the other half use "Next" and "Previous" keys. Here, the best strategy for the Logout-Action template is to loop between Screen templates that enable the user to press the "Previous" key, and Screen templates that enable the user to press the "Exit" key.

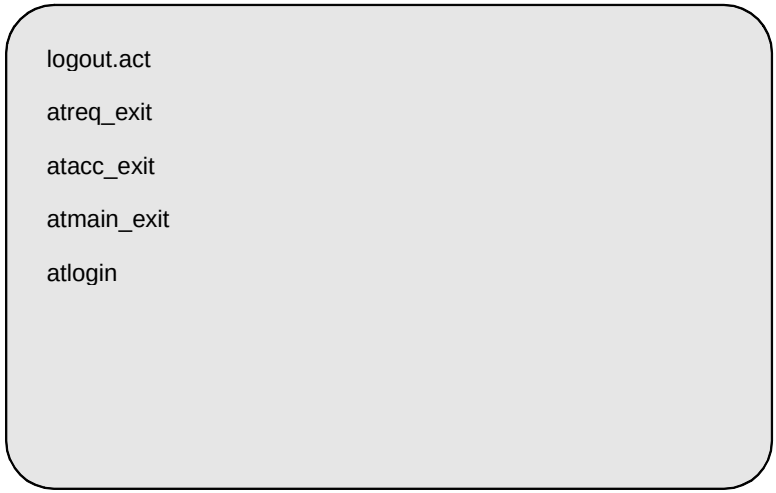
> ***Note 1:*** If you are creating applications for the TN3270, include transactions in your Logout-Action template that enable your system to recover successfully if your Telnet server goes down.
>
> ***Note 2:*** Nortel recommends that you include transactions in your Logout-Action template that enable the MRS to recover successfully if your host communications server is brought down and back up again. The sample template in Figure 6-9 does not have this capability. Compare Figure 6-10, which enables the MRS to recover.

Sample Logout-Action template

Entering a hyphen (-) for the logout-action entry indicates that a Logout-Action template is not specified. If neither a Reset-Action template nor a Logout-Action template is specified and the transaction fails, then the host computer application remains at the point where the transaction failed. Subsequent transactions that attempt to use this session also experience errors because the screen where the session remained would not be the expected starting screen, unless the transaction can start from any screen.

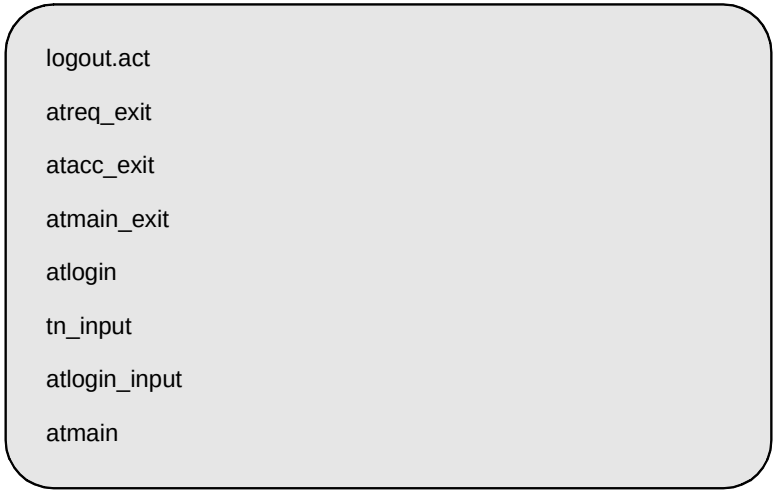
When you create a Logout-Action template, you do not specify Reset-Action or Logout-Action templates. The following figure shows a sample Logout-Action template for Figure 6-7.

Figure 6-9: Sample Logout-Action template



However, as shown in Figure 6-10, you can include transactions in your Logout-Action template that enable the MRS to recover successfully if your host communications server is brought down and back up again.

Figure 6-10: Sample Logout-Action template enabling host communications server recovery



Keep-Alive or Heartbeat Action template

You specify the heartbeat in the format `actiontemplate@`*n*, where *n* is the number of seconds between each execution of the Action template when the session is idle.

The Heartbeat Action template uses the same syntax as the Action templates described earlier in this chapter, and usually specifies only a single Screen template. That Screen template is usually the last screen specified in the Initial-Action template. Typically, your Screen template includes only a key-descriptor line, usually the <Enter> key and validation tag.

Creating Screen templates

A Screen template tells the host link exactly how to maneuver through one specific host screen. A Screen template can perform several actions on the host screen as it maneuvers through the screen. These actions include the following:

- locating landmarks on the screen to verify that the Screen template is on the correct screen
- locating information to return to your Symposium OPEN IVR application
- sending information from your Symposium OPEN IVR application to the host application
- entering keystrokes and navigational commands, just as a terminal operator enters these keystrokes and commands

By using an Action template to specify the order of Screen templates, you provide the host link with all the information it needs to navigate through a transaction from start to finish.

Define one Screen template for each host screen. Each host screen contains fields, or areas on the screen where text appears. A field may be writable, meaning the host application expects information to be entered at this point. A field may be a region where the host application displays data, or a location where static characters are displayed. For example, the header on the screen is considered a field.

A Screen template can use a field on the host screen for several purposes:

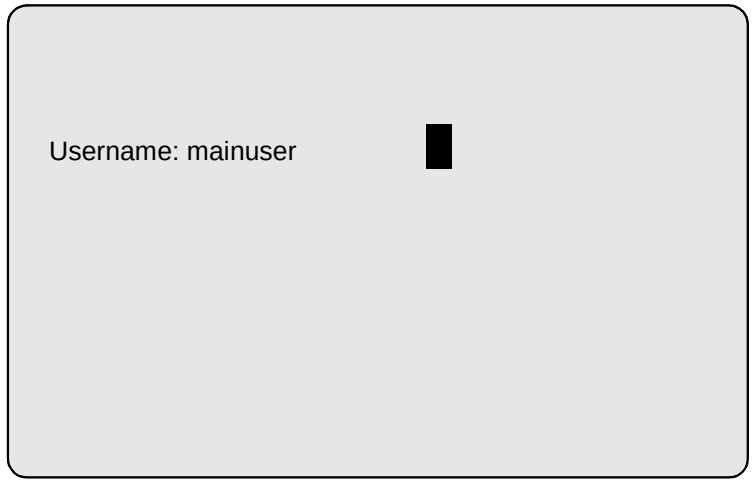
- as a reference point or navigational aid that allows the Screen template to verify that it is on the correct screen
- as a target for information sent from your Symposium OPEN IVR application
- as a source of information to return to your Symposium OPEN IVR application

A transaction consists of entering specific data in certain fields, retrieving data from other fields, and pressing keystrokes. The Screen template locates all the fields on the screen that the host link uses to process a transaction. Include in the template only the fields that are used in the transaction.

You give the Screen template the location of the fields on the host screen by specifying them in `row, column` format in the field descriptors.

For example, a Screen template can perform the following steps in order to maneuver through the login screen shown in this figure.

Figure 6-11: Fields on the host screen



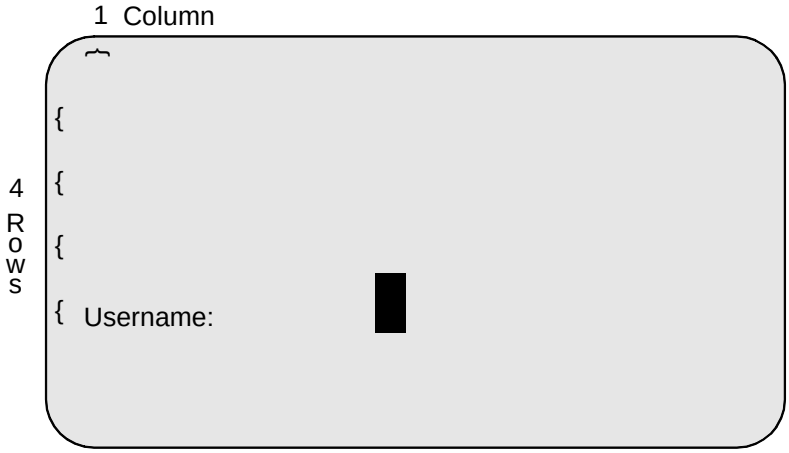
1 Each Screen template first validates that it is on the correct screen. Therefore, ensure that each Screen template checks for a validation tag that uniquely distinguishes this screen from all other host screens.

If host screens do not contain a unique identifier, use the tag that is the most unique. Always define a validation tag.

In Figure 6-11, the word `Username` provides the validation tag, since it appears only on this particular screen. You specify the location of the validation tag in `row, column` format. Locate the first character of the validation tag. From the upper-left corner of the screen, count down the number of rows, then over the number of columns to this character.

2 The Screen template then locates the first space in the field. As with locating the validation tag, the coordinates you type represent the first space in the field. Figures 6-12 and 6-13 illustrate how to determine the coordinates of a text entry field:

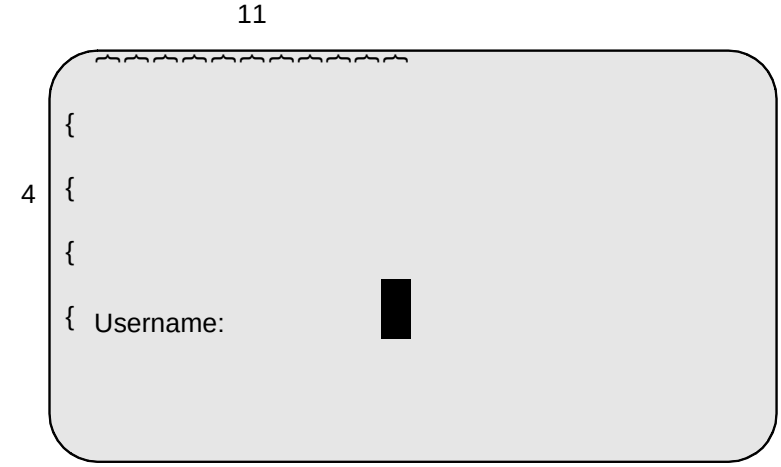
Figure 6-12: Row and column locations (part 1)



In this example, the row and column location of the validation tag corresponds to the `U` in `Username`. Specify **`4,1`** for the location of the validation tag.

Note: When determining the coordinates of a text entry field, take into account any punctuation and spaces after the prompt.

Figure 6-13: Row and column locations (part 2)



3 The Screen template enters the text **`mainuser`** in this field. Once the Screen template locates the text entry field, it transmits characters into the field. These characters can come from your Symposium OPEN IVR application, or you can code the information directly into your Screen template. These characters comprise the second part of the field descriptor.

The cursor after the `Username` prompt represents the first character space in the field. You want to tell your Screen template to use this location to begin to enter information. The coordinates of the cursor are `4,11`. This information comprises the first part of the field descriptor. In your actual Screen template, you combine steps 2 and 3 in the field descriptor.

At this point, your Screen template can return information to your Symposium OPEN IVR application in the same way that data is transmitted to the host application. This procedure is described in detail in “Screen template syntax” on page 6-28.

4 The Screen template has the user press the <Enter> key. Once the Screen template enters its information in the text entry field or reads its information, it indicates that a key needs to be pressed to tell the host application to move to the next screen. This portion of the Screen template is known as the key descriptor. The key descriptor does not need associated location coordinates.

5 The Screen template waits for a specified length of time for the next screen to display. Sometimes the host does not display the next screen instantly. This portion of the Screen template is known as the sleep descriptor.

The exact syntax of each of these descriptors is described in the following section.

Screen template syntax

Screen templates are ASCII text files you create with a text editor. Ensure that you adhere to the following guidelines:

- Screen templates reside in the `/u/nortel/3270` or `/u/nortel/5250` directory.
- Use the filename extension `.scn`

For example, if you create a Screen template called `EnterName.scn`, and you are running 3270 software, the template uses the following path:

```
/u/nortel/3270/EnterName.scn
```

The following example shows the syntax of a Screen template:

```
#comment
screen_name validation_tag_offset validation_tag
screen_descriptor
screen_descriptor
```

where *`comment`* is the comment, *`screen_name`* is the name of the screen, *`validation_tag_offset`* is the validation tag offset, and *`validation_tag`* is the validation tag.

This section provides information about the following Screen template components:

- `#comment`
- `screen_name`
- `validation_tag_offset`
- `validation_tag`

#comment

Comment lines start with the `#` symbol, which you can embed anywhere in the Screen template. A comment takes the entire line. Non-comment fields cannot precede or follow the comment on that line.

screen_name

The first non-comment line specifies the name of the Screen template. This is the Screen template file name without the `.scn` extension.

validation_tag_offset

This entry specifies the row and column of the validation tag.

There are two additional special uses of `validation_tag_offset 0,0`:

- You can type **`0,0`** for the validation tag and a hyphen (-) as the validation tag to validate the screen. For example, to execute a command at the system prompt, you do not need to verify your location at a specific screen, if you ensure that the screen contains a system prompt.

  Note: You should validate screens whenever possible. This ensures that data for the host computer application is sent to the correct screen and data sent back to the Symposium OPEN IVR application is from the correct screen.

- You can also type **`0,0`** to tell the host link to search the screen for the validation tag if you do not know the exact location of the validation tag or if the location of the validation varies. Type **`0,0`** for the offset, and then type the validation tag in the appropriate field.

validation_tag

The validation tag is an identifier on the host screen that uniquely identifies the screen to the host link. It enables the Screen template to verify that it is on the correct screen. If the Screen template cannot find its validation tag, the Screen template's governing Action template executes the Logout-Action template.

The text entry appears in the same location every time and is unique to a screen. In the example presented in this section, `Username` will always display at location `4,1` whenever the login screen is displayed.

If the screen you need to validate is a blank screen, you type the following line for screen_name, validation_tag_offset, and validation_tag:

`blank 1,1 BLANKSCREEN`

where the word *`blank`* is a placeholder, `and 1,1` and `BLANKSCREEN` are required fields.

Screen descriptor syntax

The host link requires the following information in order to enter information into or retrieve information from a field:

- the location of the field
- the description of the action to perform on the field

You provide this information in the screen descriptor. You can define multiple screen descriptors in one Screen template, but you should use one Screen template for each host application screen.

The following figure shows the syntax for the three types of screen descriptors:

- field descriptor
- key descriptor
- sleep descriptor

Figure 6-14: Screen descriptor syntax

```
# Field descriptor
position  field-name field-action

# Key descriptor
position > keyname

#Sleep descriptor
position @ sleep time
```

Note: The order of the field-descriptor lines in your Screen template file determines how data is passed to the host by the COMI cell and retrieved from the host by the COMO cell. Chapter 4, "Installing and configuring EXPRESS TN3270 or TN5250", describes how to use Symposium OPEN IVR cells to retrieve data from the host computer. The order of the field-descriptor lines in your Screen template file determines how data is passed to the host by the COMI cell and retrieved from the host by COMO cell. Chapter 7, "Using the COM cells to access the host computer", describes how to use Symposium OPEN IVR cells to retrieve data from the host computer.

position

This entry indicates the location of the field on a terminal screen. Type the exact location of the field in `row, column` format.

If you do not know the exact location of the field the TRS is reading from, or if the location of the field varies, you can type **`0,0`** and the TRS searches for a match to the name specified in *`field_name`*. For unformatted, character-based applications, specify the exact location.

In SNA transactions, writing to the terminal screen occurs at the current cursor position. Therefore, if the field I/O action is writing text to the host screen, the TRS writes the text to the screen at the current cursor position, regardless of the row and column you specify.

If you type **`0,0`** and a hyphen (**`-`**) for the `row, column` and *`field_name`* entries, the field I/O specified is executed at the current cursor location.

The position for key descriptors and sleep descriptors is always `0,0`.

field_name

This entry identifies the field that the host link accesses on the terminal screen. If the field descriptor is `0,0`, the field name uniquely matches the text on the screen or is a hyphen to indicate that the field is at the current cursor position. When entering the field name, the host link right-justifies all field names and removes all extra white space. If the transaction fails, this field identifies the current screen so that the Reset-Action template is executed.

If you enter **`0,0`** and a hyphen (**`-`**) for the `row, column` and *`field_name`* entries, the specified I/O field is executed at the current cursor location.

To specify a location on the screen that does not have an associated field name, use a hyphen (`-`) for the field name entry. If you use a hyphen to specify a field name entry, specify a location, such as `24,8`, for the location of the action specified by the I/O field entry.

field_action

This entry indicates the action to be taken on the field. The following table explains the valid entries for this field.

Entry	Description
*	Transfers the contents of the next Symposium OPEN IVR input buffer to the indicated field.
$num	Transfers the contents of the field into the next Symposium OPEN IVR output buffer. The number portion of the entry num is a number in the range from 1 to 31 and represents the number of characters returned from the host. If you do not assign a value to num, the host link places 31 characters into the buffer.
text	Places a text string in the field. If any of the special characters listed in this table are included in the text, enclose the entire text in quotation marks (for example, "$abc").
BLANKSCREEN	Clears host link memory space associated with the host application screen before retrieving output.
" "	If text contains one of the following signs —, *, $, %, >, or @ — enclose them in quotation marks. Place the signs *****, **$**, **%**, **>**, or @ in quotation marks if your field descriptors require their use without their associated buffer commands.
%	Indicates that the text that follows is an internal variable. Internal variables are used within the Screen templates only. They are not passed to the host application.
>	Indicates a key descriptor; a key name follows.

Entry	Description
@	Indicates a sleep descriptor; a sleep time follows. When entering a sleep descriptor, separate the entries on the line by white space or tab characters. If you enter text for the field action, the host link ignores any white space you include with the value until it encounters the new line character (for example, the value includes several words).

If the application running on the host is stream-based, meaning the screen scrolls as the user enters data and retrieves responses, type the field descriptor as follows:

```
0,0  -  BLANKSCREEN
```

In this instance, the field descriptor clears host link memory space associated with the application screen so that your application call flow can identify where to retrieve the appropriate data.

Searching a screen using the field descriptor

If you need to enter information into or retrieve information from a field that does not appear in a consistent location on the screen, the field descriptor can search for the field. To do this, type the following:

- **0,0** for the location
- the word or characters located immediately to the left of the writable field
- the field action command

The Screen template searches the screen for the word you specify and then performs the action command in the next writable field.

keyname

This is the name of the key for this screen.

The following table lists valid keys.

Key names		
ATTENTION	FORWARDWORD	PF11
BACKSPACE	HOME	PF12
BACKTAB	INSERTCHAR	PF13
BACKWWORD	NEWLINE	PF14
CLEAR	PA1	PF15
CURSORDOWN	PA2	PF16
CURSORLEFT	PA3	PF17
CURSORLEFTDBL	PF1	PF18
CURSORRIGHT	PF2	PF19
CURSORRIGHTDBL	PF3	PF20
CURSORUP	PF4	PF21
DELCHAR	PF5	PF22
DUP	PF6	PF23
ENTER	PF7	PF24
ERASEEOF	PF8	RESET
ERASEINPUT	PF9	SYSREQUEST
FIELDMARK	PF10	TAB

sleep_time

This entry specifies the number of seconds for the session to wait. For example, to wait 25 seconds, code the sleep descriptor as follows:

```
0,0 @ 25
```

Sample Screen template

The following figure shows an example of a Screen template file.

Figure 6-15: Screen template example

```
#Screen template file to obtain the current balance;
 Customer 1,1 Account
 #places balance into buffer

 0,0 Balance: $

 0,0 > ENTER

 0,0 @ 5
```

In this example, the second line describes the Screen template file. The screen name is `Customer`, the name of the Screen template file without the `.scn` extension. `1,1` represents the location of the validation tag on the screen. The row is listed first, followed by the column. `Account` is the screen validation tag.

The third line is a comment that describes what to do with the balance.

The fourth line is the field descriptor that describes an action to take. This field descriptor is going to find an exact match to `Balance:` and place the contents of the field into a buffer. The field descriptor line varies, depending on the action for a field. For example, you can type the last line in the example as follows:

```
2, 48  -   $
```

This example places the contents of the field starting at `2,48` into the next buffer.

Testing your Action and Screen templates

When you finish creating your Action and Screen templates, test them to ensure that the syntax is correct.

You test your Action and Screen templates with `trs -c`.

Using trs -c

Perform the following steps once you have created your Action and Screen templates and placed them in the `/u/nortel/3270` or `/u/nortel/5250` directory.

1 Change the current directory to `/u/nortel/exe` by typing the following command:

`cd /u/nortel/exe`

Press <Return>.

2 Type the following command:

`./trs -c`

Press <Return>.

The system displays the following messages:

```
TRS Checking Tools........Version 4.00.00
 TRS was compiled for......RS6000
 Total Channel Allocation..12
 Total screen templates....23
 Total action templates....14
 SNA Based Product........System Strategies
 Communication Board(s)....1
 Total defined sessions....16
 Session mapping...........No
 The Screen Templates......OK
 The Action Templates......OK
 appl name:[ssi]
 BD:0 SS:002-017 protocol: SNA
```

```
Init action[ssiInit],idle_action[]
Runtime configuration.....OK
```

If an Action or Screen template contains a syntax problem, `trs -c` indicates which template contains the problem and specifies the problem. The `trs -c` procedure also checks for the presence and syntax of the additional data files required by the host link.

Checking the trs.conf file with trs -c

If the `trs.conf` file is not created or does not reside in the `/u/nortel/3270` or `/u/nortel/5250` directory, then the output from `trs -c` is as follows:

```
TRS Checking Tools........Version 4.00.00
trs.conf file not defined
```

If `trs.conf` contains a syntax error, the output from `trs -c` contains the following:

```
ERR: Unable to load configuration file
```

If `trs.conf` cannot access the specified Initial-Action template, a syntax error registers and you get this message.

Checking the trs.sdf file with trs -c

If you did not create the `trs.sdf` file, or if the file does not reside in `/u/nortel/3270` or `/u/nortel/5250`, then the output from `trs -c` contains the following:

```
trs.sdf file not defined
Running as a Dummy Process
```

Checking Screen templates with trs -c

If there is a syntax error within a Screen template, the output from `trs -c` contains lines similar to the following:

```
ERR: No header data for Screen file
../3270/clear.scn
Running as a Dummy Process
ERR: Parse string xxxxx of screen clear
Running as a Dummy Process
```

The text after `ERR:` describes the error and then indicates which Screen template contains the error. In this example, the Screen template containing the error is `clear.scn`. The first error message indicates that there is a syntax error in the header of the Screen template `clear.scn`. The second error message indicates that `trs -c` encountered a problem trying to parse the string `xxxxx` in the Screen template `clear.scn`.

Checking Action templates with trs -c

If an Action template contains a syntax error, the output from `trs -c` contains lines similar to the following:

```
ERR: read head data from action file
../3270/exit.act
Running as a Dummy Process
ERR: Screen clear of Action exit not found
Running as a Dummy Process
```

The text after `ERR:` describes the error and then indicates which Action template contains the error. The first error message indicates that there is a syntax error in the header of the Action template `exit.act`. The `trs -c` also checks to make sure that the Screen templates listed in the Action template actually exist in the `/u/nortel/3270` or `/u/nortel/5250` directory. The second error message indicates that `trs -c` could not find the Screen template `clear.scn` listed in the Action template `exit.act`.

Using the TRS log file

This section provides information on how to debug SNA templates using `trs.log`. The TRS does not create a log unless the `startup` file is modified to put the TRS in verbose mode. To place the TRS in verbose mode, modify the `/u/nortel/startup` file. On the `trs` line, add the `-v` flag as follows:

```
./trs -b -v
```

Restart the Symposium OPEN IVR software and SNA software to create the log. The log is stored in the `/u/nortel/3270` or `/u/nortel/5250` directory as the `trs.log`. This log shows all data sent to and received from the host.

Use the `trs.log` file to develop templates or to debug a problem. The format of the `trs.log` file is system-dependent, because most of what can be used to search through the log consists of parts of the host screens. For example, if a company has its customers' phone numbers on each screen, this information can be used as a search criterion to locate transactions.

All Symposium OPEN IVR calls are listed by channel number, but the `trs.log` file lists transactions according to the logical unit (LU) number. At the beginning of a transaction, lines in the log indicate the channel and LU numbers. Use these lines to determine the LU that the transaction is using. For example:

```
14:09:35: CH=001 BD0SS005 Parse:Action_buffer:Billing
```

This example shows that channel 1 is conducting a transaction on LU 5. If the channel number is 999, then no channel maintains control of the LU. When pings are issued, they are conducted with a channel of 999 because it is the TRS conducting the ping, not a caller.

To find a transaction, you can conduct a variety of searches. By using the name of the Action template without the `.act` extension, you find the start of the transaction. Searching for the number of the channel you were on in this format, `CH=xxx` finds the start of the transaction, too. Searching for a string that is sent in the COMI finds the buffers being sent in to the transaction. Searching for a part of a host screen can find the body of that screen.

Once a part of a transaction is found, the transaction can be followed step by step until its end. First, find the LU number of the transaction, in the format of `BD0SS005,` where the number on the end is the LU number. By searching on this, you can find most of the lines that are part of the transaction.

As you step through the transaction, you see different actions occurring. Here are some of these actions in order:

```
14:09:35: CH=001 BD0SS005
Parse:Action_buffer:Billing
```

This shows the Action template that was called from the COMI.

```
CH=001 BD0SS005 * Read Input Screen BODY BEGIN *

(Blank Screen)
```

This shows how the host screen appears. In this case, it is blank.

```
14:09:35: Validate_scn: Valid Tag is BLANKSCREEN
```

This shows the name of the Screen template.

```
14:09:35: Expected screen is: sCustomerInfo

14:09:35: Checking the BLANK screen
```

This shows the valid tag that the Screen template is looking for input to screen.

```
14:09:35: BD0SS005 STR=Bill

14:09:35: BD0SS005 STR=J

14:09:35: BD0SS005 STR=Smith
```

This shows strings sent from the COMI; the Screen template is sending them to the screen.

```
14:09:35: Writing Bill (len= 4) to P.S 1

14:09:35: SNA: BD0SS005 Copy String to PS Succeeded

14:09:35: Writing J (len= 1) to P.S 5

14:09:35: SNA: BD0SS005 Copy String to PS Succeeded

14:09:35: Writing Smith (len= 5) to P.S 10

14:09:35: SNA: BD0SS005 Copy String to PS Succeeded
```

These are the strings written to the screen.

```
14:09:35: BD0SS005 <==Write Screen Done
14:09:35: BD0SS005 <==Write Screen to the Host is
done **
Username: Bill J Smith
14:09:35: BD0SS005 <==Send with AID key ENTER
```

These are the strings sent to the screen. An <Enter> key is sent. Once the <Enter> key is sent, you must wait for the next screen to return. To find the next action, search for an LU number, such as `BD0SS005`.

```
CH=001 BD0SS005 * Read Update Screen BODY BEGIN *
User Bill J Smith
Account Balance: $100
Amount Due: $25
Due Date: 120197
```

This shows the next screen from the host.

```
14:09:37: Hold: CH=001 Reply Code=0 with 8 buffers
BUF #0 =25
BUF #1 =120197
```

Chapter 7: Using the COM cells to access the host computer

This chapter explains how to use the COM cells to develop a Symposium OPEN IVR host communications application that processes one or more terminal sessions. It also contains a sample application. This chapter includes the following sections, which you may find useful as you are developing an application:

- Overview of the application development process
- Setting the COMI cell parameters
- Setting the COMO cell parameters
- Using the COMA cell
- Determining successful COM cell communication
- Sample application

Overview of the application development process

Begin the application development process as follows:

- Determine the telephone interaction with the caller, and create the corresponding Symposium OPEN IVR call flow.
- At the point in the call flow where you require interaction with the host computer, insert a COMI cell.
- Insert a COMO cell to receive the output from the mainframe host.

  Note: Always follow a COMI cell with a COMO cell, even if you do not require the return of any data. Connect error branches to a COMA cell to resolve any potential problems.

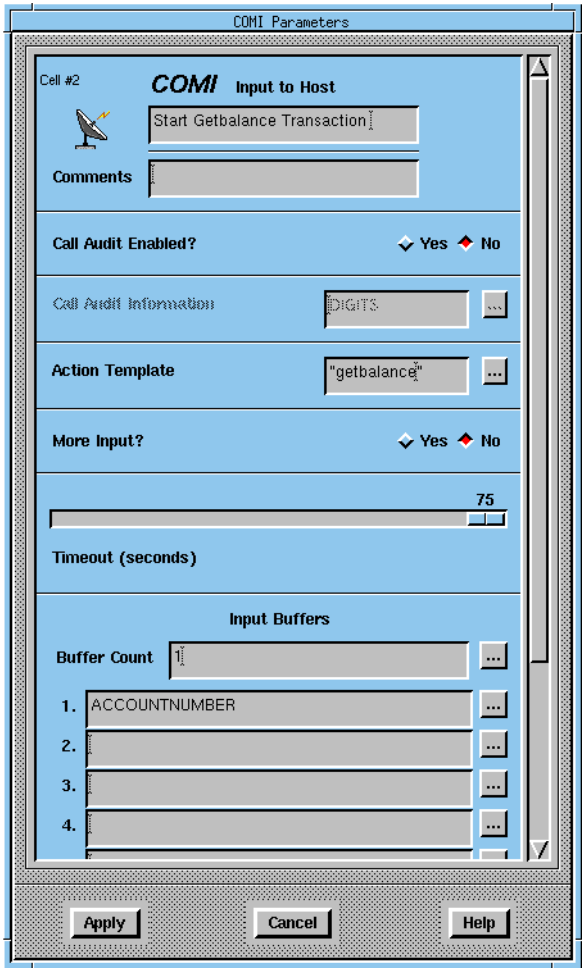
Setting the COMI cell parameters

The COMI cell starts a transaction with the host. This cell invokes an Action template which contains the commands needed for the communication transaction.

COMI cell parameters page

The following figure shows the COMI cell parameters window. This section explains how to use this cell.

Figure 7-1: COMI cell parameters window.



The following sections describe what you enter for each COMI cell parameter field. Refer to the *Cell Catalog* for additional information about the COMI cell.

Cell name

The cell name identifies the function of the cell. For example, the cell name shown in Figure 7-1 on page 7-3 is `Start Getbalance Transaction`.

Comments

In this space, type any information about the COMI cell.

Action template

Type the name of the Action template in quotes without the file extension `.act`. The name can also be a buffer containing the name of the action template. Figure 7-1 on page 7-3 shows `"getbalance"` as the name.

More Input?

If you need to send more than 10 input buffers to the host computer, click the `Yes` button; if you send 10 or fewer input buffers, click the `No` button. If you specify `Yes`, place additional COMI cells on the drawing board until you have a sufficient number of input buffers. Connect the success branch of the first COMI cell to the input branch of the following COMI cell, and then connect the rest of the COMI cells in this manner. Select the `No` button for the `More Input?` parameter in the last COMI cell in the application.

Timeout

Select the number of seconds you want the TRS process to wait for the transaction to complete by moving the slider under `Timeout`. You can move the slider from 1 to 75 seconds. This selection indicates the maximum time for the COMI cell to start the transaction and send the necessary input buffers, and for the COMO cell to receive the output buffers. The TRS process rejects the transaction and sends back a timeout if the action is not performed within the time allotted.

> ***Note:*** The application takes the timeout branch of the COMO cell if a timeout occurs.

Buffer count

Leave this cell blank as Symposium OPEN IVR automatically calculates the number of buffers when you click the Apply button.

Input buffers

Enter the names of the input buffers containing the information to pass to the TRS process. For example, the input buffer shown in Figure 7-1 on page 7-3 is ACCOUNTNUMBER (without any quotes). In this example, you program your application to place the customer's account number in the ACCOUNTNUMBER buffer before the COMI cell executes. ACCOUNTNUMBER is a user-defined buffer; you can use any name for your applications, or you can use system buffers.

> ***Note:*** You enter the input buffers in the order in which they are used by the host computer.

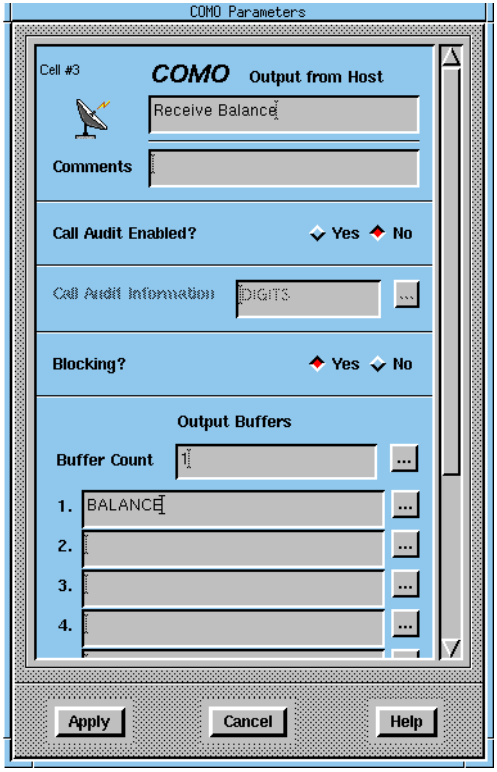
Setting the COMO cell parameters

After you set the COMI cell to send information, you code the COMO cell to receive information. Place a COMO cell directly after the COMI cell to complete the transaction, even if the host computer is not sending any data to an output buffer. The TRS process sends either verification that the transaction completed successfully or notification that there was an error in the COMO cell.

COMO cell parameters page

Figure 7-2 shows the COMO cell parameters window. This section explains how to use this cell.

Figure 7-2: COMO cell parameter window



The following sections describe what you should enter for each COMO parameter field.

Cell name

The cell name `Receive Balance` shown in Figure 7-2 on page 7-6 identifies the function of the cell.

Comments

In this space, you type any information about the COMO cell.

Blocking?

Activating blocking tells the COMO cell to wait for the transaction to complete before continuing to the next cell. The application waits until the TRS interaction ends before it continues through the Symposium OPEN IVR application (following the `END OF DATA` or `MORE DATA` branch; only one of these cells should have a branch). If you select `No`, the COMO cell receives a status code from the TRS process. Refer to the COMO cell branches section that follows.

Buffer count

Leave this field blank, as Symposium OPEN IVR automatically calculates the number of buffers when you click the `Apply` button.

Output buffers

Type the name of the buffers that accept the output from the host computer. Figure 7-2 on page 7-6 shows `BALANCE`, with no quotes, as the only output buffer for this example.

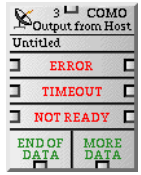
> ***Note:*** You enter the output buffers in the same order that the TRS process sends them back.

If your application uses more than 10 output buffers, string COMO cells together until you have enough buffers. Connect the input branch of each COMO cell to the `MORE DATA` branch of the previous COMO cell.

COMO cell branches

As shown in Figure 7-3, the COMO cell contains several branches.

Figure 7-3: COMO Cell



The following table describes each of the COMO cell branches.

Branch	Reason taken
ERROR	Any problem with any part, the host transaction causes the application to take the error branch.
TIMEOUT	The transaction took more than its allowed time as specified in the COMI cell.
NOT READY	The transaction did not complete, and Blocking was set to No.
END OF DATA	The transaction successfully completed.
MORE DATA	The transaction requires additional output buffers.

For each COMO cell, use either the END OF DATA branch or the MORE DATA branch, but not both. If you use the MORE DATA branch, the next cell must be another COMO cell. The last COMO cell should use the END OF DATA branch.

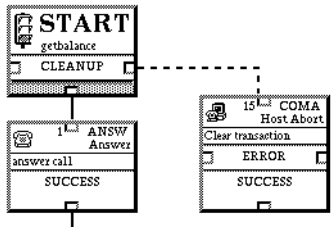
Using the COMA cell

If you selected manual mode in the application template, you use the COMA cell to abort a transaction in progress and release the session. For example, you can use a COMA cell in the clean-up branch of the START cell to deal with an error, such as a caller hanging up in the middle of a transaction.

COMA cell parameters window branches

The COMA cell frees all memory and buffers associated with the COMI and COMO cells. Figure 7-4 provides an example of a COMA cell window. The cell parameters contain a cell name field and a comments field. In this example, the COMA cell is named `Clear transaction`.

Figure 7-4: COMA cell in the clean-up handler



Determining successful COM cell communication

A System Activity Monitor (SAM) trace returns specific codes that allow you to determine if the cells communicated with the host. The following table lists these codes.

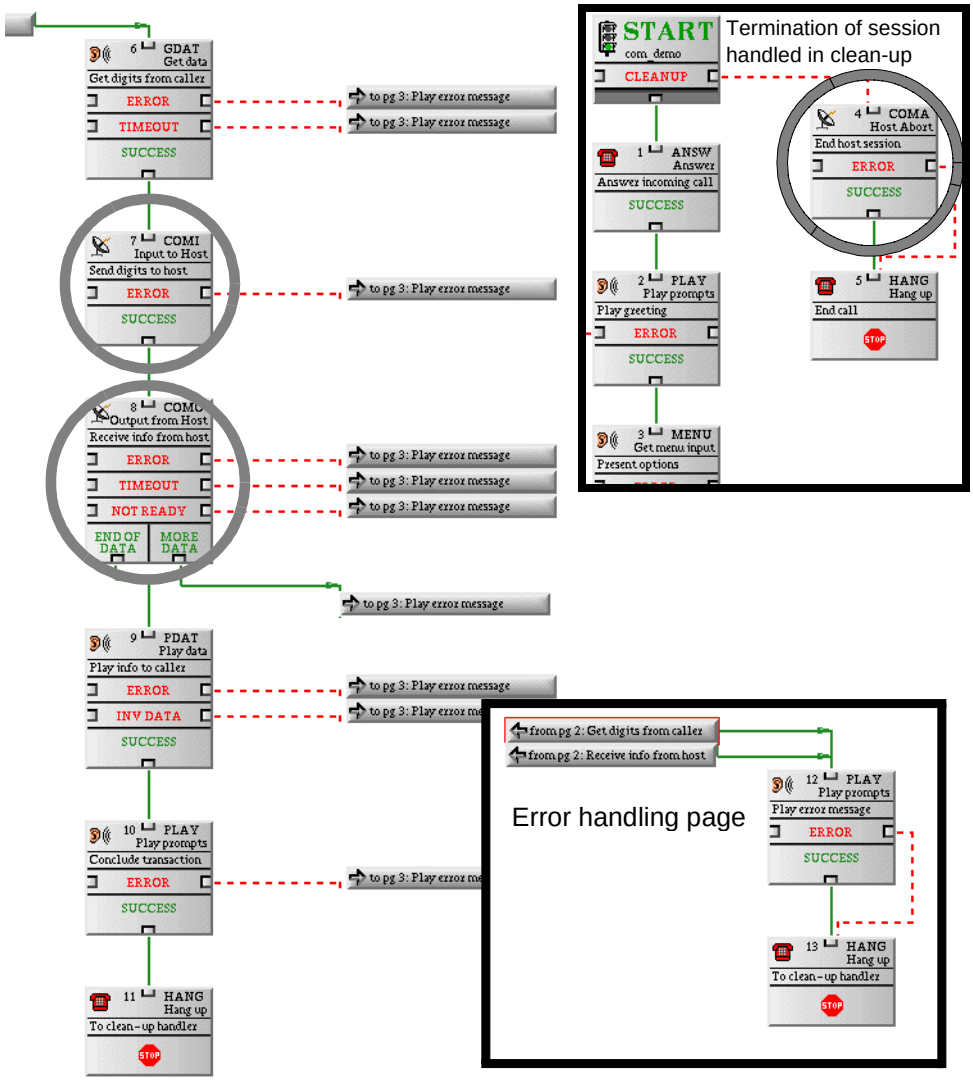
Return code	Description
0	Completed transaction; returned up to 10 buffers to the Symposium OPEN IVR application.
1	The transaction failed; memory resource error or a command sequence failure.
2	The transaction failed; process failure.
3	The transaction failed; process timeout.
4	Not used.
5	The transaction received a `Not Ready` response from the host link.
6-7	Not used.
8	Transaction succeeded; more input is required.
9	Transaction succeeded; more output is required.

Sample application

Figure 7-5 on page 7-11 shows a sample Symposium OPEN IVR application that uses the COMI, COMO, and COMA cells to retrieve a customer's account balance stored on a host computer. A caller activates this application by pressing **1** after hearing the prompt played by the `MENU` cell. A separate action template and its associated screen templates need to be created to support menu choice 2.

To illustrate the logic of the call flow, these applications are shown as one-page applications with no error branches. These applications span several pages and connect all error branches appropriately.

Figure 7-5: Symposium OPEN IVR application accessing the TRS process from the COMI, COMO cells



Before this application is executed, the Initial-Action template is executed when the Symposium OPEN IVR software is started on the application processor. Once a call from a customer is received, the Symposium OPEN IVR application performs the steps listed in the following table.

Cell number	Function
Cell 1	The ANSW cell answers the incoming call.
Cell 2	The PLAY cell plays greeting and introductory prompts to the caller.
Cell 3	The MENU cell plays a prompt instructing the caller to press **1** to get an account balance, or to press **2** to obtain an account update. In this example, the caller presses **2** for an account update.
Cell 4	The COMA cell is executed after the caller hangs up. This terminates the host session.
Cell 5	The call ends.
Cell 6	When the caller presses **2** on the telephone keypad, the GDAT cell plays the prompt requesting the caller to enter an account number. The number is stored in the ACCOUNTNUMBER buffer.
Cell 7	The COMI cell starts the transaction. The Action template specified in the COMI cell parameter window is "Getbalance". The COMI cell provides the contents of the ACCOUNTNUMBER buffer to the host computer application.
Cell 8	The COMO cell receives the customer's balance from the BALANCE buffer.
Cell 9	The PDAT cell plays the contents of the BALANCE buffer to the caller.
Cell 10	The PLAY cell plays prompts to conclude the call.
Cell 11	The caller hangs up, and the application moves to the cleanup handler (cells 4 and 5).

Cell number	Function
Cell 12	In the event of an error in the main call flow, this PLAY cell informs the caller that an error has occurred.
Cell 13	The call moves to the cleanup handler (cells 4 and 5).

If the caller hangs up before the transaction is completed, the COMA cell in the clean-up branch of the START cell clears the memory and buffers associated with the application.

Chapter 8: Troubleshooting

This chapter discusses the tools available for gathering and viewing debugging information.

Debugging the EXPRESS application

EXPRESS software provides three sets of tools that gather and view information for debugging EXPRESS:

- log file
- trace file
- information from verbose processes
- TRS audit tool

The log file and trace file components are integrated into both the character-based user interface and graphical-based user interface. You modify the `express_adm` script to invoke EXPRESS processes in verbose mode.

The EXPRESS product also provides information on the status of the connection between host and application processor.

Log file

The log file records events, errors, and abnormal conditions in an ASCII file. Multiple ASCII files (segments) comprise a log. You can define the maximum number of events that the file can contain. When the maximum number of events is reached, the existing file is closed and a new file is started.

The type of status can include all activations and deactivations, detected recoverable conditions (XID or INACTIVITY timeouts), and any non-recoverable conditions (sense code information) detected by the background processes.

This file provides the basic functionality and configuration of the different classes of events. You can save and print the log files.

For more information on the Log file, refer to the *EXPRESS Administrator's Guide*.

Trace file

The Trace file captures data traffic at either the link or station levels on the current configuration.

Data traffic information includes ACTPU, ACTLU, BIND, and UNBIND information. This file provides the basic functionality and configuration of the different classes of events, and it allows you to save and print the log files.

For more information on the trace file, refer to the *EXPRESS Administrator's Guide*.

Background process debugging

Background process debugging allows you to view the information received at each background process. You activate this type of debugging by modifying the `express_adm` script. Specific instructions are included in the script itself.

Use this type of information carefully, since it creates and outputs information to a boundless ASCII file. When you use this process, terminate the data collection as soon as you acquire the necessary information.

Traceable processes

The following table provides information on the processes that you can race, the default output file, and the type of information generated.

Process	Default input file	Information
cdmserver	**cdmserver.out**	Records database read/write operation.
executive	**executive.out**	Records event messages and requests for resources.
agent	**agent.out**	Records log messages from kernel SNA communication startup through XID negotiations.
watchdog	**watchdog.out**	Records 3270/EHLLAPI messages.
jobex	**jobex.out**	Records RJE related operations.
tn3270d	**tn3270d.out**	Records EXPRESS tn3270 Server messages.

Debugging levels

Each process supports different levels of debugging capability. The following table lists the debugging levels and information available at the different levels.

Debug level	Information presented
0	No debugging.
1	Basic debugging.
2	Inter-process messages.
3	Messages from the kernel.
5	Reserved.
7	Most verbose level of debugging, also provides the most information. Use this debugger only when requested by the vendor.

Debugging the TRS application

You can use the following tools to debug a TRS application:

- Run TRS in verbose mode.
- Use the audit tool for dynamic logging.

Refer to “Terminal Resource Server (TRS) Messages” in the *Symposium OPEN IVR Error Messages Guide* for information about any message that you see.

Running TRS in verbose mode

Use the switch `-v` to start up TRS in verbose mode. The output is recorded in the `trs.log` file in the `/u/nortel/3270` or `/u/nortel/5750` directory.

TRS audit tool

When TRS is the target process, the audit tool sets flags on it. Some of these flags display information; others require the audit process to remain active in order to gather information from the target process. The audit tool supports three additional switches that are specific to the TRS process, as shown in the following table.

Switch	Purpose
-s <log size>	Changes the number of records TRS allows in its log file before wrapping back to the top.
-L <lu range>	Identifies all the logical units to be logged.
-m <message type>	Lists the message types to log where the variable message type is a list of any of the following types: VITAL, WARNINNG, APPL, INFO, or SCREEN.

Using the audit tool for dynamic logging

You can use the audit tool to dynamically turn logging on and off, and to specify parameters to limit the amount of information that is logged. You can use the standard features of the audit tool to turn logging on and off, and to change the file name or location. Refer to the *EXPRESS System Administration Guide* for more information about using the audit tool. In addition, TRS provides three user commands to further define the logging parameters:

- message type switch
- changing log file size
- specifying logical units

Using the message type switch

You can add the `-m <message type>` switch to the audit command. This switch specifies all the message levels that will be logged. The following table lists the TRS message levels.

Level	Message type	Meaning
0	`STARTUP`	Initial startup status
1	`VITAL`	Serious errors
2	`WARNING`	Non-fatal errors
4	`APPL`	Application errors
8	`INFO`	Normal call flow
16	`SCREEN`	Screen Images

Changing log file size

You can change the number of records in the log file by using the `-s <log size>` switch. The default size is 50 000 records.

Specifying logical units

You can use the `-L <lu>` switch to define ranges of LUs to track. You can enter a list of ranges, and only the LUs you specify are included in the log.

Troubleshooting Action templates

Occasionally, something in your application may not function properly. A host-application formatting inconsistency might make a line on the screen appear in a different location, for example, or an unexpected screen may appear. These types of problems affect Action templates.

To prepare for these situations, create a Logout-Action template that can return a failed transaction to a point where the Initial-Action template can take over, and attempt to repeat the transaction.

Troubleshooting host communications

This section contains some common host communications problems and the suggested actions you can take.

After installation, EXPRESS does not come up.	
Meaning	If you did the installation as `root`, `root` is not an EXPRESS user.
Action to Take	Either switch to user `express` (`express -u express`), or define `root` as an EXPRESS user.
At startup time, `trs.log` reports that no sessions are active.	
Meaning	The vendor communication initialization failed.
Action to Take	1 Stop the Symposium OPEN IVR software. 2 Restart the vendor software. 3 Restart the Symposium OPEN IVR software. For information on starting and stopping Symposium OPEN IVR, refer to the *System Administration Guide.* For information on starting and stopping EXPRESS, refer to the template file `vtekhost_adm` or `vtekhost_adm.tn`, depending on the EXPRESS software version you are using.

Extended errors appear in `trs.log` after startup.	
Meaning	These messages are sent from the vendor software to TRS to signal an error occurring at the lower levels. Create and gather a vendor log file. Refer to “Debugging the EXPRESS application” on page 8-1 for information about debugging EXPRESS applications and creating and gathering a vendor log file.
Action to Take	Create and gather a vendor log file. For more information about creating and gathering a vendor log file, refer to “Debugging the EXPRESS application” on page 8-1.
A logical unit that has been successfully logged on is now consistently reporting a status of LOCKED.	
Meaning	Usually, when an LU locks, an application cannot verify all possible error cases and continues to function as if no error occurred. By continuing to send data, the host buffer fills, which causes the LU to lock.
Action to Take	Verify that your application screen templates check for a unique key on each screen. Also, verify that the host responded to the `ping` action.
The host link enters a logout loop after reconnection.	
Meaning	The screen(s) generated by the host was not expected by the application.
Action to Take	Stop and then restart all components (Symposium OPEN IVR, TRS, and vendor software). Include this unexpected screen in your Logout-Action template for the next time.
Scripts that autostart EXPRESS sessions do not execute properly or only execute by using the `sh` command as a prefix.	
Meaning	The execute bit is not set on the file.
Action to Take	Use `chmod` to set the executable bit.

Host reconnection is not occurring.	
Meaning	The vendor software was not configured to keep retrying reattachment.
Action to Take	Ensure that the vendor software is configured to keep trying reattachment. Use one of the provided scripts to force the vendor software to attempt reattachment.
When I have more than 64 logical units defined, I get an error message indicating all EXPRESS users greater than 64 cannot start.	
Meaning	You need to change the limits for the maximum number of support logical units.
Action to Take	Use the EXPRESS Manager to change the limits for the maximum number of supported logical units. You access this parameter by selecting the `Users/Session Configuration` option from the `Terminal Emulation Services` menu. Refer to the section about setting limits for terminal emulation in Chapter 3, “Installing and configuring EXPRESS 3270/5250 SNA/SDLC”, for additional information about setting limits for terminal emulation.

Appendix A: Planning your configuration

Planning your Local Configuration

The following information is required when you plan your host connection. You must know the number of links you need, and the type of links that are required.Your application processor can support either SDLC or Token Ring connections to your host computer.

Number of logical units (LUs) needed		
Number of stations needed		
Link information		
SDLC	Number of links	
	XID block number	
	XID ID number	
	Station address(es)	
Token Ring	Number of links	
	Local MAC address	
	Local SAP address	

Planning your Remote Configuration

Retrieve the following information for each link definition on the host computer. Typically, you obtain this information from the VTAM configuration macros.

NETID		_____
DIAL	no = leased	_____
	yes = switched	_____
DUPLEX	full = RTS is constant	_____
	half = RTS is toggled	_____
NRZI	yes = NRZI	_____
	no = NRZ	_____

SDLC station (PU) definitions

ADDR	Two-digit hex station address	_____
IDBLK	Send XID block	_____
IDNUM	XID ID	_____
MAXDATA	Maximum frame size	_____
DLOGMOD	Logmode table name that contains BIND information	_____
LOCADDR	LU address(es) of your local LUs	

Planning your Remote Configuration (*continued*)

Token Ring link definitions

NETID	Network ID	
SSCPNAME	Control point name	
CUADDR	Host's MAC address	
SAPDDR	Host's SAP address	

Token Ring station (PU) definitions

IDBLK	Send XID block	
IDNUM	XID ID	
LANSDWDW	Transmit window size	
LANACK	Number of frames before acknowledgment	
MAXDATA	Maximum frame size	
DLOGMOD	Logmode table name that contains BIND information	
LOCADDR	LU address(es) of your local LUs	

Appendix B: Sample group host configuration

1) From root, type . /etc/profile

2) express -u express -r

3) Server Configuration

4) Define Communications

5) SNA

6) Highlight Node, Alt F, open

Type	Name	Description
Node	UniverseGr	Control Point LU

SNA Node UniverseGr

- Network ID UNIVERSE
- Control Point ATRI827
- Process ..... nortel.kernel

- XID
  - XID Format
  - check Format 3

- don't check Format 0
- XID Block 012
- XID ID 00827

- RTM - Response Time Monitoring
  - check Supported
  - check Initially Enabled
  - check Host (under control)
  - don't check Local (under control)
  - check Counter overflow (under Unsolicited Support)
  - check Session End (under Unsolicited Support)
  - check First Character (under Definition)
  - don't check Last Character (under Definition)
  - don't check Keyboard Unlocked (under Definition)
  - don't check Change Direction or End Bracket (under Definition)
  - Boundaries (in seconds)
    - 1 .... 1.0
    - 2 .... 2.0
    - 3 .... 5.0
    - 4 .... 10.0

- Alert Generation
  - check Automatic Alert Generation
  - don't check Single Host Focal Point
  - check Related Host Focal Point
  - leave blank Host Focal Point Link Station

Type	Name	Description
LU	UniverseGr	Control Point LU

Node UniverseGr

LU 6.2 Using All Stations

Network Qualified Name: Network ID ... UNIVERSE

LU Name ATRI827

Address 0

Computer System ..... (blank)

User ID ..... express

Overall LU-LU Session Limit 2048

IP Address ..... (blank)

Type	Name	Description
Link	TRLink	Token Ring Link

Node UniverseGr

Local MAC Address

- check Use Default
- don’t check Assign Address

- port nortel.tr
- don’t check Connection Network Link

- MODE
    - Receive Window 33
    - Transmit Window 33
    - Maximum Frame Size 3849
    - Number Frames for Acknowledgment 1
    - Number Frames to Accept while Busy 0

    - don't check Dynamic Windowing Supported
    - check Quit When Reach Retry Limit
    - don't check Send FRMR If Unsolicited Final Bit
    - check Send FRMR If Bad Send Sequence Number

- CLASS
    - Class of Service (blank)

- TIMERS
    - Timers (in seconds, immediate or forever)
    - REJ 2.0
    - Remote Busy 10.0
    - Inactivity ..... forever
    - XID negotiation ..... forever
    - Response 2.0
    - Acknowledgment 0.1
    - Retries (value, none, or unlimited)
    - XID Attempts 40
    - Response Limit 30

TG CHARACTERISTICS

- Security
    - check Non Secure
    - don't check Public Switched Network
    - don't check Underground Cable
    - don't check Secure Conduit
    - don't check Guarded Conduit
    - don't check Encrypted
    - don't check Guarded Radiation

- Propagation Delay
    - don't check Minimum
    - check Local Area Network
    - don't check Telephone
    - don't check Packet Switched Network
    - don't check Satellite
    - don't check Maximum

- Ring Speed
    - don't check 4 Mbps
    - check 16 Mbps

- Cost Per Connect Time 128
- Cost Per Byte 128

- User Defined
  - 1 128
  - 2 128
  - 3 128

Type	Name	Description
LU	TRStation	Token

Node UniverseGR

Link TRLink

Connection Type

- check Host
- don’t check Peer-to-Peer

- Send XID
- Block 012
- ID 00827

- don’t check Remote Node is Network Node
- check Restart connection if it fails

- Local SAP Value 1C
- Remote SAP Value 04
- Local MAC Address ..... 0000J29BJD32J (can’t be changed here)
- Remote MAC Address 400052552873

- Remote Node Name
- Network ID ..... (blank)
- Control Point ..... (blank)

    - XID
        - XID Format
            - check Format 3
            - don't check Format 0
        - Role
            - don't check Secondary
            - don't check Primary
            - check Negotiable
            - don't check Verify Adjacent Node Identity
    - Priority
        - check Use Default Priority
        - don't check Assign Priority

Type	Name	Description
Station	LU2	Logical Unit 2

Node	UniverseGR
Link	TRLink
Station	TRStation

LU Addresses 2

- Setup
  - don't check Terminate on Disconnect
  - Send Window 7

****

**** UNDER SERVER CONFIG -> Define Users/Session ****

****

Type	Name	Description
All Users	All	Configuration: Express
Group	sysadm	System Admin Group
User	express	Default User of EXPRESS
Session	a	Session a

- Defined for User express
- Type: Display
- check Displayable Task
- don't check Graphic Supported
- LU/Host LU2
- Language ..... US
- Long Name ..... (blank)
- Model 2
  - Printing
  - Initial Print Mode
    - check No Printing
    - don't check Print to a disk file
    - don't check Write to print spooler
    - don't check Pipe to command
    - don't check Send to Printer Unit
    - Printer Profile Standard

- Initself
- Initself Mode
- check Disable
- don’t check Initiate Only
- don’t check Initiate and Enqueue

• IBM Mode Name	• IBM Mode Name	(blank)
• Destination LU Name	• Destination LU Name	(blank)
• User Data	• User Data	(blank)

- EHLLAPI Support
- Keystroke Intercept Queue Size 100
- Host Event Queue Size 50
- Number of Host Event/Close Intercept Queues ... 1
- Maximum Direct Session Data Size 2048
- Number of Direct Session Data Buffers 1

- Initial Session Mode
- check Intercept 3270 data stream
- check Automatic Acknowledgment (SNA only)
- don’t check Process DO Structured Fields
- don’t check Local Print Intercept
- don’t check Transparent Mode (BiSync Only)
- don’t check Validate Only Command Codes (Bisync Only)

Glossary

This section lists terms and definitions related to the information presented in this guide.

3270 protocol

Set of communication rules used by workstations designed for use with IBM mainframe computers.

5250 protocol

A set of communication rules used by terminals designed specifically for use with application system computers, such as AS/400, as well as earlier IBM system computers (36 and 38).

ACTLU

SNA command code used to activate the logical unit.

ACTPU

SNA command code used to activate the physical unit.

AID

Attention identifier; the key that interrupts the operation of the host to process the current terminal input.

application

With respect to Symposium OPEN IVR, an application is a program that controls the activity on one or more telephone trunks connected to an application processor (AP). With respect to a host computer, it is any type of program that carries out a task.

application processor

A computer or workstation running Symposium OPEN IVR.

Application Programming Interface (API)

Software libraries that enable the development of applications in a new environment, which communicate with applications in the legacy system.

bind

SNA command code used to initiate a session.

branch

A pathway between cells in a Symposium OPEN IVR application.

caller

A person whose phone call is received or originated by a Symposium OPEN IVR application.

cell

The basic element of a Symposium OPEN IVR application. Each cell performs an action, such as playing a prompt to a caller. Each cell has a set of branches to other cells. After the cell performs its action, it determines which branch the application should follow to the next cell.

channel

A telephone trunk within a cluster of MRSs.

COMA cell

Symposium OPEN IVR cell that aborts a transaction with an SNA host application.

COMI cell

Symposium OPEN IVR cell that sends input to an SNA host application using the TRS process.

COMO cell

Symposium OPEN IVR cell that receives output from an SNA host application through the TRS process.

Control Point name

The Control Point name uniquely identifies the node within the particular SNA network. The name must be unique within the network, or APPN (5250) directory searches do not work properly.

CP

Control Point

Data link

Data links connect an SNA node to other adjacent nodes in the network. A data link is connected to a computer system through a physical port on a communications adapter.

EHLLAPI

Extended High Level Language Application Programming Interface allows you to search through the host screens for specific data and conduct actions on the data. This is also called screen scraping.

emulation

Imitating a computer or computer system with a combination of hardware and software. Enables programs written for one computer to be run on another.

host

A networked computer that provides applications and services to other networked computers. In this guide, a host is an IBM computer that supports the SNA communication protocol.

link

A physical connection between two devices.

link station

A link station represents a logical connection between two endpoints over a physical data link, such as an SDLC link or a Token Ring Local Area Network. On any SDLC logical connection between two stations, one is the primary station and the other is the secondary station. The role of each side may be preconfigured to be primary or secondary, or may be defined as negotiable, in which case the role is determined at connect time. For SDLC stations, the role is configured in the Data Link definition. For each link station, the secondary station address must be configured. This is the address assigned to the secondary station and is contained in all frames transmitted on the connection. The Token Ring data link represents a physical connection to a Token Ring LAN. For each configured link station, the following are specified: (1) Local Service Access Point (SAP) address, (2) Remote SAP address, and (3) the Remote MAC address. Most commonly, the local and remote SAP addresses are 04. The remote MAC address must be configured to match the address of the remote node.

LU (logical unit)

A logical unit (LU) is a port through which a user accesses the SNA network to communicate with other users and gain access to network resources. It is typically an SNA terminal or print device. LUs are configured under link stations in the Express software.

LU6.2-based file transfer

A protocol for high-performance cooperative processing, specifically as a file transfer mechanism. The cornerstone of LU6.2 is the concept of a pair of transaction programs (TCP/IP and SNA); file transfer and print sharing are implemented with companion programs, one on each target system.

MRS

Multimedia resource server

network name

An SNA node is uniquely identified in the network by a Control Point (CP) name. An SNA network is also identified by a unique name, called the Network ID. Nodes and LUs are addressable entities in an SNA network. The names of the CPs and LUs within a single SNA network must be unique. The names of network addressable units are comprised of two parts: the network name, followed by a period, followed by the LU (or CP) name. Each of the two elements that make up the network qualified name can be 1 to 8 characters long, so that the fully qualified name can be up to 17 characters long (including the period).

node

A grouping composed of an application processor connected to one or more MRSs.

remote node name

When configuring a connection to an adjacent node, the `Remote Node Name` parameter is used to identify the remote node. This parameter consists of two parts: the Network ID and the CP Name. If this is a host connection being used for 3270 or LU 0 sessions, the `Remote CP Name` parameter is not required, unless adjacent node identity verification or incoming connections require it. If the connection is being used for LU6.2 sessions (5250), then the setting of the remote CP name is very important.

screen scraping

See EHLLAPI.

sense code

SNA error code returned by the IBM host application.

session

3 A connection between two systems for the exchange of user data. Sessions are between logical units (LUs).

4 A connection to a host as defined in the `trs.conf` file, representing a terminal connection.

SNA (Systems Network Architecture)

A group of IBM protocols that define how IBM computers (and any other computer that uses the SNA protocol) transmit and receive data. This protocol is also used by many other computer hardware and software manufacturers. It presumes an underlying transport such as SDLC, Token Ring, Ethernet, or X.25.

station

The physical device to which you communicate.

TCP/IP (Transmission Control Protocol/Internet Protocol)

Originally developed for U.S. military communication, this protocol has become a de facto standard worldwide and the protocol of choice for backbone routing of data traffic. It presumes an underlying communications capability such as Token Ring or Ethernet.

templates

ASCII files used by the TRS process to manage the SNA terminal session.

terminal emulation

Allows users in a non-IBM environment to log in to the applications running in the legacy (IBM) environment just as if they had legacy environment terminals (3270 and 5250).

transaction

The function performed by a set of Action and Screen template files when executed by the TRS.

TRS (Terminal Resource Server)

The Symposium OPEN IVR process that manages communication between an IBM host and the Symposium OPEN IVR application processor.

unbind

SNA command code used to terminate a session.

validation tag

This is an identifier on the host screen that uniquely identifies the screen to the host link.

validation tag offset

This entry in the screen template specifies the row and column of the validation tag.

XID (eXchange Identification)

When two nodes first establish communications, they exchange identification information in data link level data units called XIDs. There are two XID formats (Format 0 and Format 3). Format 3 is the recommended format. The XID contains a 32 bit value called an exchange identifier. The first 12 bits are the block ID and typically identifies the IBM product that is sending the XID. AS/400 systems use 0x056 as the block ID. Express uses 0xe05 as the block ID. The next 20 bits represent the node ID. Some systems, such as host nodes, use this ID, rather than the CP name, to identify the partner node during XID negotiation. If the system is connecting to a host node, then the configured block ID and node ID must match the value configured at the Host.

Index

Symbols

#comment, 6-13

Numerics

3270/5250
  SNA/SDLC card, 2-2
  Token Ring card, 2-9
9-Pin connector pinouts, 2-10

A

Action templates, 6-2, 6-3, 6-4, 6-5, 6-7, 6-8, 6-17
  checking with trs -c, 6-39
  creating, 6-8
  Heartbeat, 5-3, 6-7
  Initial-Action, 6-6, 6-8, 6-16
    sample, 6-17
  Keep-Alive, 6-7
  Logout-Action, 6-7, 6-20
  Reset-Action, 6-7, 6-18
    sample, 6-20
  syntax, 6-8, 6-13
  troubleshooting, 8-8
  types of, 6-3, 6-6
action_name, 6-14
AP software host communication cells, 1-6
app_name, 6-14
application development
  process, overview, 7-2
  voice, 6-2
architecture, flow, 1-5
audit tool, TRS, 8-5

B

background
  process debugging, 8-2
  starting user groups, 4-17
beginning EXPRESS SNA software installation, 3-8
bringing up
  interactive session, 4-14
    for a particular user, 4-15
  Symposium OPEN IVR, 5-7

C

changing log file size, 8-6
channels
  mapping to host communications sessions, 5-10
checking
  status of links, 3-22
  status of stations, 3-22
  trs.conf file with trs -c, 6-38
COM cells
  for host communications, 7-1, 7-10
COMA, 1-7, 7-9
  cell parameters, 7-9

return codes, 7-10
using, 7-9
COMI, 1-7, 6-31
cell parameters, 7-3
return codes, 7-10
setting, 7-2
comment, 6-29
communication terms, SNA, 2-2
communications
adapter, 1-9, 3-4
hardware, installing, 3-4
media, 3-3
COMO, 1-7, 6-31
cell branches, 7-8
cell parameters, 7-6
return codes, 7-10
configuring
completing, 4-14
/etc/rc2.d/S99zgen file, 3-31
EXPRESS, 3-16, 4-8
EXPRESS TN3270/TN5250, 4-1
group host sample, B-1
host communications, 5-8
host link, 4-1
OPEN IVR, 5-1
planning, A-1
sample configurations, 5-10
session files, 4-8
specifying login IDs, 5-6
specifying passwords, 5-6
trs.sdf file, 5-5
connector and connector pin assignments, D-25, 2-5
creating
Action templates, 6-8
Screen templates, 6-24
Symposium OPEN IVR applications that communicate with an SNA host, 1-5
template files, 6-1

D

D-25 connector, 2-5
daemons
starting up, 3-25
DB-15
connector, 2-4
loopback connector wiring diagram, 2-6
debugging
background process, 8-2
EXPRESS application, 8-1
levels, 8-4
TRS application, 8-5
defining
links, 3-16, 3-17
logical units, 3-16
LU 6.2 (5250 only), 3-18
LU pools (3270 only), 3-18
mode, 3-20
remote LUs, 3-19
session groups, 4-8
sessions, 3-20, 4-8
stations, 3-16, 3-17
user groups, 4-8
users, 3-20, 4-8
displaying
interactive session, 3-24
dynamic logging
using the audit tool for, 8-6

E

editing system startup files, 3-30, 4-17
EHLLAPI, 1-9
emulation characteristics
IBM 3270, 2-2
IBM 5250, 2-2
establishing a naming convention, 3-16
/etc/rc2.d/S99zgen file, 3-31
EXPRESS
configuring, 3-16, 4-8

daemons and users
starting up, 3-25
hardware, 1-9
installation
files modified, 3-7
PS process
starting, 4-7
software, 1-9
installing, 3-6
removing earlier versions, 3-8
EXPRESS SNA
installing, 3-9
Server, 3-1, 3-30
checking links and stations, 3-22
configuring, 3-16
debugging, 8-1
defining a node, 3-17
defining users and sessions, 3-20
editing system startup files, 3-30
starting, 3-15
software
beginning installation, 3-8
EXPRESS SNA Server
installing, 3-6
EXPRESS TN3270/TN5250, 4-1
configuring, 4-1, 4-8
debugging, 8-1
defining users and sessions, 4-8
editing system startup files, 4-17
installing, 4-1, 4-3
software
installing, 4-3
starting, 4-7
terminal emulation, 4-5
verifying connections, 4-14

F

field, 6-24
descriptor, searching a screen using, 6-34
writable, 6-24
field_action, 6-33
field_name, 6-32
field-descriptor syntax, 6-31
files
editing startup, 3-30
/etc/rc2.d/S99zgen, 3-31
log, 8-1, 8-2
changing size, 8-6
lubuf.dat, 5-6
modified by EXPRESS installation, 3-7
startup
editing, 4-17
syntax
notes, 5-2
trace, 8-1
trs.conf, 5-2, 6-5
syntax, 5-2
trs.sdf, 5-5
checking with trs -c, 6-38
setting up, 5-5
verbose mode, 8-1

G

group host configuration, sample, B-1

H

hardware, 1-3
EXPRESS, 1-9
host
application, sample, 6-9
connectivity, overview, 2-1
information, 3-3, 4-2
link, 1-8
architecture, 1-5
communications, 1-1

sessions, mapping channels to, 5-10
verifying connections, 3-22
host communications
application, 1-1
cells, AP software, 1-6
configuring, 5-1, 5-8
installing, 4-1
links, 1-1
sessions
mapping channels to, 5-10
starting session, 4-14
testing 3270/5250, 3-23
troubleshooting, 8-8
using COM cells, 7-1

I

IBM
3270 emulation characteristics, 2-2
5250 emulation characteristics, 2-2
application, 1-8
protocol, 1-9
screens, 1-8
ID for LU, 5-6
Initial-Action templates, 6-8, 6-16
sample, 6-17
See also Action templates
installing
communications hardware, 3-4
EXPRESS SNA, 3-9
EXPRESS software, 3-6
EXPRESS TN3270/TN5250, 4-1, 4-3
hardware, 3-4
host link, 4-1
Madge Token Ring card, 2-11
SNA hardware, 3-5
SYNC/570 adapter, 2-7
interactive session
displaying, 3-24

K

keyname, 6-35

L

links
checking status, 3-22
defining, 3-16, 3-17
host communications, 1-1
local configuration, planning, A-1
logical units
defining, 3-16, 3-18
specifying, 8-7
specifying login IDs, 5-6
specifying passwords, 5-6
login ID, specifying, 5-6
logout_action, 6-14
Logout-Action templates
sample, 6-21
See also Action templates
LU
6.2 (5250 only),defining, 3-18
pair, 3-19
pools (3270 only), defining, 3-18
specifying IDs, 5-6
specifying passwords, 5-6
lubuf.dat file, 5-6

M

Madge Smart 16/4 AT Plus Ringnode
token ring host card, 2-9
Madge Token Ring card
installing, 2-11
manual mode, 6-15
mapping channels to host sessions, 5-10
message type switch, using, 8-6
messages, host communication, 8-8

mode
manual, 6-15
verbose, 8-1
modifying
/etc/rc2.d/S99zgen file, 3-31

N

naming convention, establishing, 3-16

O

operating system, 2-1

P

package contents, verifying, 3-3, 4-2
parameters page
COMI cell, 7-3
COMO cell, 7-6
parameters window branches, COMA cell, 7-9
password
for LU, 5-6
password, specifying, 5-6
physical configuration, 4-2
requirements, reviewing, 3-3
pin assignments, 3270/5250 SNA/SDLC card, 2-4
planning your local configuration, A-1
position, 6-32
process debugging, background, 8-2

R

remote LUs, defining, 3-19
removing earlier versions of the EXPRESS software, 3-8
requirements, system, 4-2
reset_action, 6-14
Reset-Action template
sample, 6-20
See Action templates, 6-7, 6-18
reviewing physical configuration requirements, 3-3
running TRS in verbose mode, 8-5

S

SAM, 7-10
sample
application, 7-10
application call flow, 1-6
configuration files, 5-10
group host configuration, B-1
host application, 6-9
Initial-Action template, 6-17
Logout-Action template, 6-21
Reset-Action template, 6-20
session file
for TN3270 communications, 4-12
for TN5250 communications, 4-9
template files, 5-10
trs.conf file, 5-3
VTAM Gen, 2-15
screen descriptor
syntax, 6-30
screen scraping, 1-9
Screen templates, 6-3, 6-5, 6-15, 6-24
checking with trs -c, 6-39
field-descriptors, 6-30
Home Screen, 6-18
moving around, 6-25
row, column, 6-25
syntax, 6-28
transparent functions, 1-9
screen_name, 6-29
SDLC, 3-4
card, 2-3
searching a screen using the field descriptor, 6-34
session
bringing up, 4-14
for a particular user, 4-15
session configuration files, 4-8

session groups
defining, 4-8
sessions, 5-3
defining, 3-20, 4-8
host communications
mapping channels to, 5-10
starting, 4-14
setting
COMI cell parameters, 7-2
COMO cell parameters, 7-5
limits for terminal emulation, 3-21, 4-5
site planning, 3-3, 4-2
requirements, 3-3, 4-2
sleep_time, 6-35
SNA, 1-1
communication terms, 2-2
hardware installation
completing, 3-5
protocols, 6-6
SNA hardware
installation, 2-7
SNA/SDLC card, 3270/5250, 2-2
software
EXPRESS, 1-9
operating system, 2-1
supported releases, 2-1
software, EXPRESS, 1-9
specifying
IDs for LUs, 5-6
logical units, 8-7
passwords for LUs, 5-6
starting
EXPRESS
daemons and users, 3-25
PS process, 4-7
TN3270 or TN5250, 4-7
EXPRESS TN3270/TN5250, 4-7
host communications sessions, 4-14
interactive session, 4-14
for a particular user, 4-15
Symposium OPEN IVR, 5-7
user groups in the background, 4-17
startup files, editing, 3-30
stations
checking status, 3-22
defining, 3-16, 3-17
status
links and stations, 3-22
switch
message type, 8-6
Symposium OPEN IVR, 6-27
call flow, 7-10
creating applications that communicate with an SNA host, 1-5
starting, 5-7
SYNC/570
adapter, installing, 2-7
card, 2-4, 2-6
syntax
Action template, 6-13
screen descriptor, 6-30
Screen templates, 6-28
understanding, 6-13
system
requirements, 4-2
startup files, editing, 4-17
System Activity Monitor, 7-10

T

template files
samples, 5-10
templates
files, 6-1
Screen, 6-3, 6-5, 6-15, 6-24
testing, 6-37
terminal
emulation
setting limits, 3-21, 4-5
terminal emulation, 1-9
EXPRESS TN3270/TN5250, 4-5
testing
3270/5250 host communications, 3-23
Action templates, 6-37
communication, 7-10
Screen templates, 6-37
status of links and stations, 3-22

TN3270
sample session file, 4-12
TN3270/TN5250, 4-1
configuring, 4-1, 4-8
debugging, 8-1
defining users and sessions, 4-8
editing system startup files, 4-17
installing, 4-1, 4-3
starting, 4-7
terminal emulation, 4-5
TN5250
sample session file, 4-9
Token Ring, 3-4
card, 3270/5250, 2-9
Pin Arrangements, 2-9
trace file, 8-2
See files, 8-1
traceable processes, 8-3
transactions, 1-7, 6-18
determining, 6-2
sample, 6-2
samples, 7-10
troubleshooting, 8-1
Action templates, 8-8
host communications, 8-8
TRS, 1-3, 1-7, 1-8
application, debugging, 8-5
audit tool, 8-1, 8-5
log, 6-39
log file, using, 6-40
modules, 1-5
startup switches, 5-7
template file, 6-1
verbose mode, running, 8-5
trs, 3-20, 4-14
trs -c
verifying trs.conf file with, 6-38
trs.conf file, 5-2
files
trs.conf, 5-2
purpose and structure, 6-5
See files, 6-5
verifying with trs -c, 6-38
trs.sdf file
configuring, 5-5
TRSC, 1-3, 1-8
TRSL, 1-8

U

user groups
defining, 4-8
starting in the background, 4-17
users
defining, 3-20, 4-8
starting up, 3-25
using
audit tool for dynamic logging, 8-6
COM cells for host communications, 7-1
COMA cell, 7-9
message type switch, 8-6
start-up switches, 5-7
trs -c, 6-37
TRS log file, 6-40

V

validation tag, 6-25
offset, 6-29
validation_tag, 6-30
verbose mode, 8-1
running TRS in, 8-5
verifying
connections to host computer, 3-22
package contents, 3-3, 4-2
required physical configuration, 4-2
status of links and stations, 3-22
trs.conf file with trs -c, 6-38
VTAM Gen, sample, 2-15

W

wiring diagram
DB-15 loopback connector, 2-6
writable field, 6-24

Symposium OPEN IVR

SNA Host Communications Guide

Toronto Information Products
Nortel
522 University Avenue, 14th Floor
Toronto, Ontario, Canada
M5G 1W7

© 1998 Northern Telecom
All rights reserved

Information is subject to change without notice. Northern Telecom reserves the right to make changes in design or components as progress in engineering and manufacturing may warrant.

Nortel, Symposium, IVR, and Meridian 1 are registered trademarks of Northern Telecom. APERTUS EXPRESS is a registered trademark of Computer Network Technology, Inc. Digi is a registered trademark of DIGI International.
Motif is a trademark of Open Software Foundation Inc. SCO is a registered trademark of The Santa Cruz Operation, Inc. UNIX is a registered trademark of Novell, Inc. Voicetek and VTK are registered trademarks of Voicetek Corporation. X Window System and X are trademarks of the Massachusetts Institute of Technology.

Product release: 4.0
Document release: Standard 1.0
Date: July 1998

Printed in the United States of America

NORTEL
NORTHERN TELECOM